Configuration

- BlueCat-DNS_DHCP-Server
- CA-Access-Gateway
- Configuration-Audit
- Configuration-Jobs
- Device-SSL-Certificates
- Events
- Instances
- Licenses
- Network-Functions
- PaloAlto-VM-Series
- SECUREMATRIX-GSB
- System
- Websense-Protector
- XenServer

BlueCat-DNS_DHCP-Server

- aclcontrol
- aclrule
- bluecat
- bluecat_device_profile
- bluecat_init_config
- bluecat_vpx_image

aclcontrol

Configuration for ACL Rule resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the system users.
start_acl_on_reboot	<Boolean>	Read-write	Tells SVM to start the ACL upon bootup or not.
rules	<String[]>	Read-write	Provides the ACL state.
state	<Boolean>	Read-write	Provides the ACL state.

Operations

(click to see Properties)

- APPLY_CHANGES
- ENABLE
- GET (ALL)
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

apply_changes

URL: https://<MGMT-IP>/nitro/v2/config/aclcontrol?action=apply_changes;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"aclcontrol": {

"id":<String_value>,
```

```
"rules":<String_value[]>,

"start_acl_on_reboot":<Boolean_value>,

"state":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aclcontrol":[{

"id":<String_value>,

"start_acl_on_reboot":<Boolean_value>,

"rules":<String_value>,

"state":<Boolean_value>}]}
```

enable

URL: https://<MGMT-IP>/nitro/v2/config/aclcontrol?action=enable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"aclcontrol": {

"id":<String_value>,

"rules":<String_value[]>,

"start_acl_on_reboot":<Boolean_value>,

"state":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aclcontrol":[{

"id":<String_value>,

"start_acl_on_reboot":<Boolean_value>,

"rules":<String_value>,

"state":<Boolean_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/aclcontrol **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/aclcontrol **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of aclcontrol resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/aclcontrol **? pagesize=#no;pageno=#no** Use this query-parameter to get the aclcontrol resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/aclcontrol **?count=yes** Use this query-parameter to get the count of aclcontrol resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aclcontrol":[{

"id":<String_value>,

"start_acl_on_reboot":<Boolean_value>,

"rules":<String_value>,

"state":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/aclcontrol/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aclcontrol":[{

"id":<String_value>,

"start_acl_on_reboot":<Boolean_value>,

"rules":<String_value>,

"state":<Boolean_value>}]}
```

disable

URL: https://<MGMT-IP>/nitro/v2/config/aclcontrol?action=disable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"aclcontrol": {

"id":<String_value>,
```

```
"rules":<String_value[]>,

"start_acl_on_reboot":<Boolean_value>,

"state":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aclcontrol":[{

"id":<String_value>,

"start_acl_on_reboot":<Boolean_value>,

"rules":<String_value>,

"state":<Boolean_value>}]}
```

aclrule

Configuration for ACL Rule resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
priority	<Integer>	Read-write	Priority. Minimum value = 1 Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
protocol	<String>	Read-write	IP Protocol. The allowed values are [TCP UDP ICMP ANY]. Minimum length = 3 Maximum length = 4
name	<String>	Read-write	Rule Name. Minimum length = 1 Maximum length = 128
dst_port	<String>	Read-write	Enable external authentication.
src_ip	<String>	Read-write	Source IP Address or Subnet. Minimum length = 3 Maximum length = 128
action	<String>	Read-write	Action can be [Allow Block]. Minimum length = 4 Maximum length = 5
id	<String>	Read-write	Id is system generated key for all the system users.
dst_ip	<String>	Read-only	Destination IP Address.
acl_status	<String>	Read-only	.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
acl_status_internal	<String>	Read-only	.

Operations

(click to see Properties)

- ENABLE
- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

enable

URL: https://<MGMT-IP>/nitro/v2/config/aclrule?action=enable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"aclrule": {

"protocol":<String_value>,

"priority":<Integer_value>,

"action":<String_value>,

"name":<String_value>,

"id":<String_value>,

"src_ip":<String_value>,

"dst_port":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aclrule":[{

"priority":<Integer_value>,

"protocol":<String_value>,

"name":<String_value>,

"dst_port":<String_value>,

"src_ip":<String_value>,

"dst_ip":<String_value>,

"acl_status":<String_value>,

"acl_status_internal":<String_value>,

"action":<String_value>,

"id":<String_value>}]}
```

add

URL: https://<MGMT-IP>/nitro/v2/config/aclrule?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"aclrule": {

<b>"priority":<Integer_value></b>,

<b>"protocol":<String_value></b>,

<b>"name":<String_value></b>,

<b>"src_ip":<String_value></b>,

<b>"action":<String_value></b>,

"id":<String_value>,

"dst_port":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aclrule":[{

"priority":<Integer_value>,

"protocol":<String_value>,

"name":<String_value>,

"dst_port":<String_value>,

"src_ip":<String_value>,

"dst_ip":<String_value>,

"acl_status":<String_value>,
```

```
"acl_status_internal":<String_value>,

"action":<String_value>,

"id":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/aclrule/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/aclrule **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/aclrule **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of aclrule resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/aclrule **? pagesize=#no;pageno=#no** Use this query-parameter to get the aclrule resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/aclrule **?count=yes** Use this query-parameter to get the count of aclrule resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aclrule":[{

"priority":<Integer_value>,

"protocol":<String_value>,

"name":<String_value>,

"dst_port":<String_value>,

"src_ip":<String_value>,

"dst_ip":<String_value>,

"acl_status":<String_value>,

"acl_status_internal":<String_value>,

"action":<String_value>,

"id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/aclrule/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aclrule":[{

"priority":<Integer_value>,

"protocol":<String_value>,

"name":<String_value>,

"dst_port":<String_value>,

"src_ip":<String_value>,

"dst_ip":<String_value>,

"acl_status":<String_value>,

"acl_status_internal":<String_value>,

"action":<String_value>,
```

```
"id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/aclrule/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"aclrule":{

<b>"id":<String_value></b>,

"protocol":<String_value>,

"priority":<Integer_value>,

"action":<String_value>,

"name":<String_value>,

"src_ip":<String_value>,

"dst_port":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aclrule":[{

"priority":<Integer_value>,

"protocol":<String_value>,

"name":<String_value>,

"dst_port":<String_value>,

"src_ip":<String_value>,

"dst_ip":<String_value>,

"acl_status":<String_value>,

"acl_status_internal":<String_value>,

"action":<String_value>,

"id":<String_value>}]}
```

disable

URL: https://<MGMT-IP>/nitro/v2/config/aclrule?action=disable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"aclrule": {

"protocol":<String_value>,

"priority":<Integer_value>,

"action":<String_value>,

"name":<String_value>,

"id":<String_value>,

"src_ip":<String_value>,

"dst_port":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aclrule":[{

"priority":<Integer_value>,

"protocol":<String_value>,

"name":<String_value>,

"dst_port":<String_value>,

"src_ip":<String_value>,

"dst_ip":<String_value>,

"acl_status":<String_value>,

"acl_status_internal":<String_value>,
```

```
"action":<String_value>,

"id":<String_value>}]}
```

bluecat

Configuration for BlueCat Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	Image Name, This parameter is used while provisioning VM Instance with XVA image, template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vm_memory_total	<Double>	Read-write	Total Memory of VM Instance in MB. 2048MB, 5120MB.
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
throughput_limit	<Double>	Read-write	Throughput Limit in Mbps set for VM Instance.
vlan_id_0_1	<Integer>	Read-write	VLAN id for the management interface 0/1. Maximum value =
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128
description	<String>	Read-write	Description of managed device. Minimum length = 1 Maximum length = 512
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
fips_partition_name	<String>	Read-write	FIPS Partition Name. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ha_ifc	<String>	Read-write	High Availability Interface.
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.
burst_priority	<Integer>	Read-write	Burst Priority of the VM Instance between 1 and 4.
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1 Maximum length = 64
service_ifc	<String>	Read-write	Service Interface.
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
is_managed	<Boolean>	Read-write	Is Managed.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device. Minimum length = 1 Maximum length = 128
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device. Minimum length = 1 Maximum length = 64
network_interfaces	<network_interface[]>	Read-write	Network Interfaces.
last_updated_time	<Double>	Read-write	Last Updated Time.
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.
host_ip_address	<String>	Read-write	Host IPAddress where VM is provisioned. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
id	<String>	Read-write	Id is system generated key for all the managed devices.
ipv6_address	<String>	Read-write	IPv6 Address.
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
license_edition	<String>	Read-write	Edition of instance.
number_of_acu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
device_family	<String>	Read-write	Device Family. Minimum length = 1 Maximum length = 64
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..
la_mgmt	<Boolean>	Read-write	Bond consisting of management ports on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_id_0_2	<Integer>	Read-write	VLAN id for the management interface 0/2. Maximum value =
number_of_ssl_cores	<Integer>	Read-write	Assign number of ssl virtual functions to VM Instance.
type	<String>	Read-write	Type of device, (Xen | NS). Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	Assign throughput in Mbps to VM Instance.
template_name	<String>	Read-write	Template Name, This parameter is used while provisioning VM Instance with template, template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
max_burst_throughput	<Double>	Read-write	Maximum burst throughput in Mbps of VM Instance.
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal network (not required to be set for SDWAN).
node_id	<String>	Read-write	Node identification of a device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1 Maximum length = 64
throughput_allocation_mode	<Integer>	Read-write	Throughput Allocation Mode: 0-Fixed, 1-Burst-able.
number_of_scu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
number_of_cores	<Integer>	Read-write	Number of cores that are assigned to VM Instance.
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			A-B, and for Cluster it will be CLIP. Minimum length = 1 Maximum length = 128
std_bw_total	<Integer>	Read-write	Standard Bandwidth.
domain_name	<String>	Read-write	Domain name of VM Device. Minimum length = 1 Maximum length = 128
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1 Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.
vlan_1_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/2 on VM Instance. Maximum value =
vrid_list_ipv4_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_1_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/7 on VM Instance. Maximum value =
vrid_list_ipv6_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV6 VMAC Generation.
vrid_list_ipv4_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV4 VMAC Generation.
vrid_list_ipv6_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV6 VMAC Generation.
if_10_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/4 on VM Instance.
vlan_10_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 10/5 on VM Instance. Maximum value =
vrid_list_ipv6_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV6 VMAC Generation.
vrid_list_ipv6_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV6 VMAC Generation.
if_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/5 on VM Instance.
receiveuntagged_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/4 on VM Instance.
vrid_list_ipv6_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV6 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV6 VMAC Generation.
if_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/3 on VM Instance.
vrid_list_ipv6_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV6 VMAC Generation.
receiveuntagged_10 _4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/4 on VM Instance.
vrid_list_ipv6_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV6 VMAC Generation.
vlan_10_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 10/6 on VM Instance. Maximum value =
if_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/5 on VM Instance.
is_new_crypto	<Boolean>	Read-write	True if number_of_acu/ number_of_scu are used, false if number_of_ssl_core s is used.
vrid_list_ipv6_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV6 VMAC Generation.
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
if_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/7 on VM Instance.
receiveuntagged_10 _8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtReceive Untagged Packets on 10/8 on VM Instance.
vrid_list_ipv6_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV6 VMAC Generation.
receiveuntagged_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/6 on VM Instance.
vlan_1_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/4 on VM Instance. Maximum value =
if_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/2 on VM Instance.
receiveuntagged_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/5 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/6 on VM Instance.
receiveuntagged_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/1 on VM Instance.
vrid_list_ipv4_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV4 VMAC Generation.
vrid_list_ipv6_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV6 VMAC Generation.
vrid_list_ipv6_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV6 VMAC Generation.
vrid_list_ipv4_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVRID List for Interface 1/7 for IPV4 VMAC Generation.
receiveuntagged_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/2 on VM Instance.
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
vrid_list_ipv4_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV4 VMAC Generation.
receiveuntagged_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/7 on VM Instance.
vlan_10_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/4 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum value =
if_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/7 on VM Instance.
receiveuntagged_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/1 on VM Instance.
vrid_list_ipv6_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV6 VMAC Generation.
if_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/1 on VM Instance.
if_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/8 on VM Instance.
l2_enabled	<Boolean>	Read-write	L2mode status of VM Instance.
vlan_10_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 10/7 on VM Instance. Maximum value =
vlan_10_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/1 on VM Instance. Maximum value =
receiveuntagged_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/2 on VM Instance.
reboot_vm_on_cpu_change	<Boolean>	Read-write	Reboot VMs on CPU change during resource allocation.
vlan_1_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/6 on VM Instance. Maximum value =
vrid_list_ipv4_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV4 VMAC Generation.
receiveuntagged_10_7	<Boolean>	Read-write	This property is deprecated by

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			network_interfaces;lt br;gtReceive Untagged Packets on 10/7 on VM Instance.
vrid_list_ipv4_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV4 VMAC Generation.
vlan_1_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/1 on VM Instance. Maximum value =
vrid_list_ipv6_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV6 VMAC Generation.
vlan_1_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/8 on VM Instance. Maximum value =
vlan_10_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 10/2 on VM Instance. Maximum value =
receiveuntagged_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/5 on VM Instance.
vrid_list_ipv4_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV4 VMAC Generation.
vrid_list_ipv4_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV4 VMAC Generation.
vrid_list_ipv4_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV4 VMAC Generation.
receiveuntagged_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			on 1/8 on VM Instance.
vrid_list_ipv4_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV4 VMAC Generation.
vlan_1_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/3 on VM Instance. Maximum value =
receiveuntagged_10 _3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/3 on VM Instance.
if_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/6 on VM Instance.
if_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/6 on VM Instance.
vlan_10_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVLAN for Network 10/3 on VM Instance. Maximum value =
vlan_1_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/5 on VM Instance. Maximum value =
if_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/3 on VM Instance.
vrid_list_ipv4_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV4 VMAC Generation.
if_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/4 on VM Instance.
vrid_list_ipv4_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/1 on VM Instance.
vrid_list_ipv4_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV4 VMAC Generation.
vlan_10_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/8 on VM Instance. Maximum value =
receiveuntagged_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/3 on VM Instance.
crypto_change_requires_reboot	<Boolean>	Read-write	True if the current changes made by user requires a reboot of the VM else false.
vrid_list_ipv4_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			IPV4 VMAC Generation.
if_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/8 on VM Instance.
vrid_list_ipv4_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV4 VMAC Generation.
vrid_list_ipv6_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for IPV6 VMAC Generation.
if_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/2 on VM Instance.
vrid_list_ipv6_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV6 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
sdxtools_http_port	<Integer>	Read-only	Http Port number to communicate to SDXTools.
instance_state	<String>	Read-only	State of device, UP only if device accessible.
reason	<String>	Read-only	Reason of failure for this managed device.
virtual_functions	<String>	Read-only	Virtual Functions assigned to VM Instance depending on SDX platform.
disk_space	<Double>	Read-only	Show Disk Space (GB) available to VM Instance.
upsince	<String>	Read-only	Upsince of managed device.
domid	<Double>	Read-only	DOM Id assigned to VM Instance viewed in hypervisor.
cpu_core_pe	<String>	Read-only	Packet Engine cores assigned to VM Instance.
vm_proteus_control	<String>	Read-only	true if BlueCat VM is under proteus control. Modify interfaces should not be allowed in that case..
tenant_id	<String>	Read-only	Tenant ID.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
preferred_cpu_socket	<String>	Read-only	Preferred CPU Socket.
device_uuid	<String>	Read-only	Device UUID.
do_config	<Boolean>	Read-only	Do default config for managed device.
version	<String>	Read-only	Device Version.
host_id	<String>	Read-only	Host ID.
has_sdxtools	<Boolean>	Read-only	True if SDX Tools are installed on this VM.
vm_memory_used	<Double>	Read-only	Memory Used (MB) by VM Instance.
number_of_cpu	<Integer>	Read-only	Number of CPU that is assigned to VM Instance.
is_swg	<Boolean>	Read-only	Boolean to indicate whether a VM is SWG.
sysid	<String>	Read-only	System ID.
vm_state	<String>	Read-only	State of Virtual Machine (Running | Halted).
uuid	<String>	Read-only	UUID of VM Instance generated by Hypervisor.
cpu_core_mgmt	<String>	Read-only	Management CPU cores assigned to VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ssl_virtual_functions	<String>	Read-only	SSL Virtual Functions assigned to VM Instance.
uptime	<String>	Read-only	Uptime of device.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.
assigned_cpu_socket	<String>	Read-only	Assigned CPU Socket 0 or 1.
bluecat_memory_usage	<Double>	Read-only	Memory Usage (%) of BlueCat Instance.
sdxtools_public_key	<String>	Read-only	SDXTools HTTPS Public Key.
status	<String>	Read-only	Status of managed device.
vm_description	<String>	Read-only	Description of the VM instance.
systemname	<String>	Read-only	Device System Name.
mac_eth0	<String>	Read-only	MAC Address of eth0 on VM Instance.
vm_rx	<Double>	Read-only	In Throughput of VM Instance in Mbps.
disk_allocation	<String>	Read-only	Disk allocation for VM Instance.
vm_tx	<Double>	Read-only	Out Throughput of VM Instance in Mbps.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
bluecat_cpu_usage	<Double>	Read-only	CPU Usage (%) of BlueCat Instance.
vm_memory_free	<Double>	Read-only	Free Memory (MB) available in VM Instance.
vm_memory_usage	<Double>	Read-only	Memory Usage (%) of VM Instance.
sdxtools_version	<String>	Read-only	SDXTools version running on VM Instance.
vm_cpu_usage	<Double>	Read-only	CPU Usage (%) of VM Instance.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- REBOOT
- STOP
- ADD
- FORCE_REBOOT
- GET (ALL)
- GET
- DELETE
- UPDATE
- FORCE_STOP
- START

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

reboot

URL: https://<MGMT-IP>/nitro/v2/config/bluecat/id_value<String>?action=reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"bluecat": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

stop

URL: https://<MGMT-IP>/nitro/v2/config/bluecat/id_value<String>?action=stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"bluecat": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

add

URL: https://<MGMT-IP>/nitro/v2/config/bluecat?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"bluecat": {

<b>"ip_address":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,
```

```
"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"service_ifc":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,
```

```
"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,
```

```
"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,
```

```
"ent_bw_available":<Integer_value>,

"description":<String_value>,

"fips_partition_name":<String_value>,

"ha_ifc":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,
```

```
"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,
```

```
"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,
```

```
"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,
```

```
"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"bluecat":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,
```

```
"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"vm_proteus_control":<String_value>,

"user_driven":<Boolean_value>,

"ha_ifc":<String_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,
```

```
"netmask":<String_value>,

"device_uuid":<String_value>,

"service_ifc":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,
```

```
"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,
```

```
"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,
```

```
"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"bluecat_memory_usage":<Double_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,
```

```
"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"bluecat_cpu_usage":<Double_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,
```

```
"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,
```

```
"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,
```

```
"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,
```

```
"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,
```

```
"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

force_reboot

URL: https://<MGMT-IP>/nitro/v2/config/bluecat/id_value<String>?action=force_reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"bluecat": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/bluecat **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/bluecat **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of bluecat resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/bluecat **? pagesize=#no;pageno=#no** Use this query-parameter to get the bluecat resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/bluecat **?count=yes** Use this query-parameter to get the count of bluecat resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"bluecat":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,
```

```
"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"vm_proteus_control":<String_value>,

"user_driven":<Boolean_value>,

"ha_ifc":<String_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"service_ifc":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,
```

```
"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,
```

```
"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>
```

```
"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,
```

```
"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"bluecat_memory_usage":<Double_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,
```

```
"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"bluecat_cpu_usage":<Double_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,
```

```
"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,
```

```
"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,
```

```
"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,
```

```
"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,
```

```
"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/bluecat/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"bluecat":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,
```

```
"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"vm_proteus_control":<String_value>,

"user_driven":<Boolean_value>,

"ha_ifc":<String_value>,
```

```
"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"service_ifc":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,
```

```
"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,
```

```
"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,
```

```
"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"bluecat_memory_usage":<Double_value>,

"throughput":<Double_value>,
```

```
"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,
```

```
"bluecat_cpu_usage":<Double_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,
```

```
"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,
```

```
"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,
```

```
"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,
```

```
"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/bluecat/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

update

URL: https://<MGMT-IP>/nitro/v2/config/bluecat/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"bluecat":{

<b>"id":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,
```

```
"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"service_ifc":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,
```

```
"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,
```

```
"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,
```

```
"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"fips_partition_name":<String_value>,

"ha_ifc":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,
```

```
"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,
```

```
"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,
```

```
"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,
```

```
"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"bluecat":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,
```

```
"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"vm_proteus_control":<String_value>,

"user_driven":<Boolean_value>,

"ha_ifc":<String_value>,

"mastools_version":<String_value>,
```

```
"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"service_ifc":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,
```

```
"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,
```

```
"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,
```

```
"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"bluecat_memory_usage":<Double_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,
```

```
"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"bluecat_cpu_usage":<Double_value>,
```

```
"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,
```

```
"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,
```

```
"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,
```

```
"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,
```

```
"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

force_stop

URL: https://<MGMT-IP>/nitro/v2/config/bluecat/id_value<String>?action=force_stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"bluecat": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

start

URL: https://<MGMT-IP>/nitro/v2/config/bluecat/id_value<String>?action=start;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"bluecat": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

bluecat_device_profile

Configuration for bluecat_device_profile resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Profile Name. Minimum length = 1 Maximum length = 128
svm_ns_comm	<String>	Read-write	Communication protocol (http or https) with Instances. Minimum length = 1 Maximum length = 10
use_global_setting_for_communication_with_ns	<Boolean>	Read-write	True, if the communication with Instance needs to be global and not device specific.
id	<String>	Read-write	Id is system generated key for all the device profiles.
type	<String>	Read-write	Profile Type, This must be with in specified supported instance types:

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			blx,sdvanvw,ns,nssdx,cbwanopt,cpx. Minimum length = 1 Maximum length = 128
ns_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix ADC profiles.
password	<String>	Read-write	Instance credentials.Password for this profile. Minimum length = 1 Maximum length = 127
snmpsecuritylevel	<String>	Read-write	SNMP v3 security level for this profile.
username	<String>	Read-write	Instance credentials.Username provided in the profile will be used to contact the instance. Minimum length = 1 Maximum length = 127
snmpauthprotocol	<String>	Read-write	SNMP v3 auth protocol for this profile.
ssh_port	<String>	Read-write	SSH port to connect to the device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
snmpprivprotocol	<String>	Read-write	SNMP v3 priv protocol for this profile.
host_password	<String>	Read-write	Host Password for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpversion	<String>	Read-write	SNMP version for this profile.
passphrase	<String>	Read-write	Passphrase with which private key is encrypted.
host_username	<String>	Read-write	Host User Name for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpsecurityname	<String>	Read-write	SNMP v3 security name for this profile. Maximum length = 31
ssl_private_key	<String>	Read-write	SSL Private Key for key based authentication.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ssl_cert	<String>	Read-write	SSL Certificate for certificate based authentication.
http_port	<Integer>	Read-write	HTTP port to connect to the device.
snmpcommunity	<String>	Read-write	SNMP community for this profile. Maximum length = 31
https_port	<Integer>	Read-write	HTTPS port to connect to the device.
max_wait_time_reboot	<String>	Read-write	Max waiting time to reboot Citrix ADC.
snmpprivpassword	<String>	Read-write	SNMP v3 priv password for this profile. Minimum length = 8 Maximum length = 31
cb_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix SD-WAN profiles.
snmpauthpassword	<String>	Read-write	SNMP v3 auth password for this profile. Minimum length = 8 Maximum length = 31

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
tenant_id	<String>	Read-only	Tenant Id of device profile.
is_default	<Boolean>	Read-only	True, if this profile is system generated and can not be deleted.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/bluecat_device_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"bluecat_device_profile": {

<b>"name":<String_value></b>,

<b>"password":<String_value></b>,

<b>"username":<String_value></b>,

"snmpsecuritylevel":<String_value>,

"svm_ns_comm":<String_value>,

"ssh_port":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,
```

```
"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"type":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"ns_profile_name":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"bluecat_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,
```

```
"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/bluecat_device_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/bluecat_device_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/bluecat_device_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of bluecat_device_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/bluecat_device_profile **? pagesize=#no;pageno=#no** Use this query-parameter to get the bluecat_device_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/bluecat_device_profile **?count=yes** Use this query-parameter to get the count of bluecat_device_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"bluecat_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,
```

```
"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,
```

```
"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/bluecat_device_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"bluecat_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,
```

```
"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,
```

```
"snmpauthpassword":<String_value>}]}
```

bluecat_init_config

Configuration for Initial Configuration Parameters of bluecat resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
prop2	<String>	Read-write	Property 2 of bluecat initial Configuration.
prop1	<String>	Read-write	Property 1 of bluecat initial Configuration.
ip_address	<String>	Read-write	IP Address of bluecat. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- ADD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/bluecat_init_config?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"bluecat_init_config": {

<b>"ip_address":<String_value></b>,
```

```
"prop2":<String_value>,

"prop1":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"bluecat_init_config":[{

"prop2":<String_value>,

"prop1":<String_value>,

"ip_address":<String_value>}]}
```

bluecat_vpx_image

Configuration for XVA images of BlueCat Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/bluecat_vpx_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/bluecat_vpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/bluecat_vpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of bluecat_vpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/bluecat_vpx_image **?pagesize=#no;pageno=#no** Use this query-parameter to get the bluecat_vpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/bluecat_vpx_image **?count=yes** Use this query-parameter to get the count of bluecat_vpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"bluecat_vpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/bluecat_vpx_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"bluecat_vpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/bluecat_vpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

CA-Access-Gateway

- ca_sps
- ca_sps_device_profile
- ca_sps_downgrade
- ca_sps_init_config
- ca_sps_upgrade
- ca_sps_vpx_image

ca_sps

Configuration for CA SiteMinder SPS Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	Image Name, This parameter is used while provisioning VM Instance with XVA image, template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.
vm_memory_total	<Double>	Read-write	Total Memory of VM Instance in MB. 2048MB, 5120MB.
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
throughput_limit	<Double>	Read-write	Throughput Limit in Mbps set for VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_id_0_1	<Integer>	Read-write	VLAN id for the management interface 0/1. Maximum value =
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128
description	<String>	Read-write	Description of managed device. Minimum length = 1 Maximum length = 512
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
fips_partition_name	<String>	Read-write	FIPS Partition Name. Minimum length = 1 Maximum length = 128
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
initial_configuration	<ca_sps_init_config>	Read-write	initial configuration parameters of ca_sps.
burst_priority	<Integer>	Read-write	Burst Priority of the VM Instance between 1 and 4.
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1 Maximum length = 64
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
is_managed	<Boolean>	Read-write	Is Managed.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 128
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device. Minimum length = 1 Maximum length = 64
network_interfaces	<network_interface[]>	Read-write	Network Interfaces.
last_updated_time	<Double>	Read-write	Last Updated Time.
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.
host_ip_address	<String>	Read-write	Host IPAddress where VM is provisioned. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key for all the managed devices.
ipv6_address	<String>	Read-write	IPv6 Address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
license_edition	<String>	Read-write	Edition of instance.
number_of_acu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
device_family	<String>	Read-write	Device Family. Minimum length = 1 Maximum length = 64
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..
la_mgmt	<Boolean>	Read-write	Bond consisting of management ports on VM Instance.
vlan_id_0_2	<Integer>	Read-write	VLAN id for the management interface 0/2. Maximum value =
number_of_ssl_cores	<Integer>	Read-write	Assign number of ssl virtual functions to VM Instance.
type	<String>	Read-write	Type of device, (Xen | NS).

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	Assign throughput in Mbps to VM Instance.
template_name	<String>	Read-write	Template Name, This parameter is used while provisioning VM Instance with template, template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64
max_burst_throughput	<Double>	Read-write	Maximum burst throughput in Mbps of VM Instance.
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			network (not required to be set for SDWAN).
node_id	<String>	Read-write	Node identification of a device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1 Maximum length = 64
throughput_allocation_mode	<Integer>	Read-write	Throughput Allocation Mode: 0-Fixed, 1-Burst-able.
number_of_scu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
number_of_cores	<Integer>	Read-write	Number of cores that are assigned to VM Instance.
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be A-B, and for Cluster it will be CLIP. Minimum length = 1 Maximum length = 128
std_bw_total	<Integer>	Read-write	Standard Bandwidth.
domain_name	<String>	Read-write	Domain name of VM Device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 128
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1 Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.
vlan_1_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/2 on VM Instance. Maximum value =
vrid_list_ipv4_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV4 VMAC Generation.
vlan_1_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/7 on VM Instance. Maximum value =
vrid_list_ipv6_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVRID List for Interface 1/2 for IPV6 VMAC Generation.
vrid_list_ipv4_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV4 VMAC Generation.
vrid_list_ipv6_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV6 VMAC Generation.
if_10_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/4 on VM Instance.
vlan_10_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/5 on VM Instance. Maximum value =
vrid_list_ipv6_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV6 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV6 VMAC Generation.
if_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/5 on VM Instance.
receiveuntagged_1_ 4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/4 on VM Instance.
vrid_list_ipv6_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV6 VMAC Generation.
vrid_list_ipv6_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV6 VMAC Generation.
if_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/3 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV6 VMAC Generation.
receiveuntagged_10 _4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/4 on VM Instance.
vrid_list_ipv6_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV6 VMAC Generation.
vlan_10_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/6 on VM Instance. Maximum value =
if_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/5 on VM Instance.
is_new_crypto	<Boolean>	Read-write	True if number_of_acu/ number_of_scu are used, false if

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			number_of_ssl_cores is used.
vrid_list_ipv6_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV6 VMAC Generation.
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
if_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/7 on VM Instance.
receiveuntagged_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/8 on VM Instance.
vrid_list_ipv6_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV6 VMAC Generation.
receiveuntagged_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtReceive Untagged Packets on 1/6 on VM Instance.
vlan_1_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/4 on VM Instance. Maximum value =
if_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/2 on VM Instance.
receiveuntagged_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/5 on VM Instance.
receiveuntagged_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/6 on VM Instance.
receiveuntagged_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			on 10/1 on VM Instance.
vrid_list_ipv4_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV4 VMAC Generation.
vrid_list_ipv6_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV6 VMAC Generation.
vrid_list_ipv6_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV6 VMAC Generation.
vrid_list_ipv4_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV4 VMAC Generation.
receiveuntagged_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/2 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
vrid_list_ipv4_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV4 VMAC Generation.
receiveuntagged_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/7 on VM Instance.
vlan_10_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/4 on VM Instance. Maximum value =
if_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/7 on VM Instance.
receiveuntagged_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Untagged Packets on 1/1 on VM Instance.
vrid_list_ipv6_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV6 VMAC Generation.
if_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/1 on VM Instance.
if_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/8 on VM Instance.
l2_enabled	<Boolean>	Read-write	L2mode status of VM Instance.
vlan_10_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/7 on VM Instance. Maximum value =
vlan_10_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/1 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/2 on VM Instance.
reboot_vm_on_cpu_change	<Boolean>	Read-write	Reboot VMs on CPU change during resource allocation.
vlan_1_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/6 on VM Instance. Maximum value =
vrid_list_ipv4_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV4 VMAC Generation.
receiveuntagged_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/7 on VM Instance.
vrid_list_ipv4_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_1_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/1 on VM Instance. Maximum value =
vrid_list_ipv6_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV6 VMAC Generation.
vlan_1_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/8 on VM Instance. Maximum value =
vlan_10_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/2 on VM Instance. Maximum value =
receiveuntagged_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/5 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv4_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV4 VMAC Generation.
vrid_list_ipv4_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV4 VMAC Generation.
vrid_list_ipv4_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV4 VMAC Generation.
receiveuntagged_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/8 on VM Instance.
vrid_list_ipv4_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV4 VMAC Generation.
vlan_1_3	<Integer>	Read-write	This property is deprecated by

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			network_interfaces;lt br;gtVLAN for Network 1/3 on VM Instance. Maximum value =
receiveuntagged_10 _3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/3 on VM Instance.
if_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/6 on VM Instance.
if_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/6 on VM Instance.
vlan_10_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/3 on VM Instance. Maximum value =
vlan_1_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/5 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/3 on VM Instance.
vrid_list_ipv4_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV4 VMAC Generation.
if_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/4 on VM Instance.
vrid_list_ipv4_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV4 VMAC Generation.
if_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/1 on VM Instance.
vrid_list_ipv4_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_10_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/8 on VM Instance. Maximum value =
receiveuntagged_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/3 on VM Instance.
crypto_change_requires_reboot	<Boolean>	Read-write	True if the current changes made by user requires a reboot of the VM else false.
vrid_list_ipv4_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for IPV4 VMAC Generation.
if_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/8 on VM Instance.
vrid_list_ipv4_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			IPV4 VMAC Generation.
vrid_list_ipv6_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for IPV6 VMAC Generation.
if_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/2 on VM Instance.
vrid_list_ipv6_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV6 VMAC Generation.
sdxtools_http_port	<Integer>	Read-only	Http Port number to communicate to SDXTools.
instance_state	<String>	Read-only	State of device, UP only if device accessible.
reason	<String>	Read-only	Reason of failure for this managed device.
virtual_functions	<String>	Read-only	Virtual Functions assigned to VM Instance depending on SDX platform.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
disk_space	<Double>	Read-only	Show Disk Space (GB) available to VM Instance.
upsince	<String>	Read-only	Upsince of managed device.
domid	<Double>	Read-only	DOM Id assigned to VM Instance viewed in hypervisor.
cpu_core_pe	<String>	Read-only	Packet Engine cores assigned to VM Instance.
tenant_id	<String>	Read-only	Tenant ID.
preferred_cpu_socket	<String>	Read-only	Preferred CPU Socket.
device_uuid	<String>	Read-only	Device UUID.
do_config	<Boolean>	Read-only	Do default config for managed device.
version	<String>	Read-only	Device Version.
host_id	<String>	Read-only	Host ID.
has_sdxtools	<Boolean>	Read-only	True if SDX Tools are installed on this VM.
vm_memory_used	<Double>	Read-only	Memory Used (MB) by VM Instance.
number_of_cpu	<Integer>	Read-only	Number of CPU that is assigned to VM Instance.
ca_sps_cpu_usage	<Double>	Read-only	CPU Usage (%) of CA SiteMinder SPS Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_swg	<Boolean>	Read-only	Boolean to indicate whether a VM is SWG.
sysid	<String>	Read-only	System ID.
vm_state	<String>	Read-only	State of Virtual Machine (Running | Halted).
uuid	<String>	Read-only	UUID of VM Instance generated by Hypervisor.
cpu_core_mgmt	<String>	Read-only	Management CPU cores assigned to VM Instance.
ssl_virtual_functions	<String>	Read-only	SSL Virtual Functions assigned to VM Instance.
uptime	<String>	Read-only	Uptime of device.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.
location	<String>	Read-only	Location of CA SiteMinder SPS Instance.
ca_sps_memory_usage	<Double>	Read-only	Memory Usage (%) of CA SiteMinder SPS Instance.
assigned_cpu_socket	<String>	Read-only	Assigned CPU Socket 0 or 1.
sdxtools_public_key	<String>	Read-only	SDXTools HTTPS Public Key.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
status	<String>	Read-only	Status of managed device.
vm_description	<String>	Read-only	Description of the VM instance.
systemname	<String>	Read-only	Device System Name.
mac_eth0	<String>	Read-only	MAC Address of eth0 on VM Instance.
vm_rx	<Double>	Read-only	In Throughput of VM Instance in Mbps.
disk_allocation	<String>	Read-only	Disk allocation for VM Instance.
vm_tx	<Double>	Read-only	Out Throughput of VM Instance in Mbps.
vm_memory_free	<Double>	Read-only	Free Memory (MB) available in VM Instance.
vm_memory_usage	<Double>	Read-only	Memory Usage (%) of VM Instance.
sdxtools_version	<String>	Read-only	SDXTools version running on VM Instance.
vm_cpu_usage	<Double>	Read-only	CPU Usage (%) of VM Instance.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- REBOOT
- STOP
- ADD
- FORCE_REBOOT
- GET (ALL)
- GET
- DELETE
- UPDATE
- FORCE_STOP
- START

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

reboot

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps/id_value<String>?action=reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ca_sps": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

stop

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps/id_value<String>?action=stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ca_sps": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

add

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ca_sps": {

<b>"ip_address":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,
```

```
"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,
```

```
"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,
```

```
"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,
```

```
"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"initial_configuration":<ca_sps_init_config_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,
```

```
"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"is_member_ifc":<Boolean_value>,

"vrid_list_ipv4":<String_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"id":<String_value>,
```

```
"l2_enabled":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,
```

```
"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"max_burst_throughput":<Double_value>,

"if_1_3":<Boolean_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,
```

```
"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ca_sps":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,
```

```
"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,
```

```
"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"initial_configuration":<ca_sps_init_config_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ca_sps_cpu_usage":<Double_value>,

"ipv4_address":<String_value>,
```

```
"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,
```

```
"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,
```

```
"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"location":<String_value>,

"license_grace_time":<Integer_value>,

"ca_sps_memory_usage":<Double_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,
```

```
"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,
```

```
"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,
```

```
"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,
```

```
"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,
```

```
"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,
```

```
"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

force_reboot

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps/id_value<String>?action=force_reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ca_sps": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ca_sps **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ca_sps resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ca_sps **? pagesize=#no;pageno=#no** Use this query-parameter to get the ca_sps resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ca_sps **?count=yes** Use this query-parameter to get the count of ca_sps resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ca_sps":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,
```

```
"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"initial_configuration":<ca_sps_init_config_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,
```

```
"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ca_sps_cpu_usage":<Double_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,
```

```
"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,
```

```
"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,
```

```
"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"location":<String_value>,

"license_grace_time":<Integer_value>,

"ca_sps_memory_usage":<Double_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,
```

```
"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,
```

```
"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,
```

```
"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,
```

```
"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,
```

```
"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,
```

```
"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ca_sps":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,
```

```
"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"initial_configuration":<ca_sps_init_config_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,
```

```
"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ca_sps_cpu_usage":<Double_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,
```

```
"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>
```

```
"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"location":<String_value>,

"license_grace_time":<Integer_value>,
```

```
"ca_sps_memory_usage":<Double_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,
```

```
"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,
```

```
"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,
```

```
"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,
```

```
"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,
```

```
"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,
```

```
"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

update

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ca_sps":{

<b>"id":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,
```

```
"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,
```

```
"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,
```

```
"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"initial_configuration":<ca_sps_init_config_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,
```

```
"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"is_member_ifc":<Boolean_value>,

"vrid_list_ipv4":<String_value>,
```

```
"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"id":<String_value>,

"l2_enabled":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,
```

```
"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"max_burst_throughput":<Double_value>,

"if_1_3":<Boolean_value>,
```

```
"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ca_sps":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,
```

```
"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"initial_configuration":<ca_sps_init_config_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,
```

```
"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ca_sps_cpu_usage":<Double_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,
```

```
"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>
```

```
"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"location":<String_value>,

"license_grace_time":<Integer_value>,
```

```
"ca_sps_memory_usage":<Double_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,
```

```
"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,
```

```
"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,
```

```
"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,
```

```
"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,
```

```
"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,
```

```
"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

force_stop

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps/id_value<String>?action=force_stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ca_sps": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

start

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps/id_value<String>?action=start;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ca_sps": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

ca_sps_device_profile

Configuration for ca_sps device profile resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Profile Name. Minimum length = 1 Maximum length = 128
svm_ns_comm	<String>	Read-write	Communication protocol (http or https) with Instances. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 10
use_global_setting_for_communication_with_ns	<Boolean>	Read-write	True, if the communication with Instance needs to be global and not device specific.
id	<String>	Read-write	Id is system generated key for all the device profiles.
type	<String>	Read-write	Profile Type, This must be with in specified supported instance types: blx,sdvanvw,ns,nssdx,cbwanopt,cpx. Minimum length = 1 Maximum length = 128
ns_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix ADC profiles.
password	<String>	Read-write	Instance credentials.Password for this profile. Minimum length = 1 Maximum length = 127
snmpsecuritylevel	<String>	Read-write	SNMP v3 security level for this profile.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
username	<String>	Read-write	Instance credentials.Username provided in the profile will be used to contact the instance. Minimum length = 1 Maximum length = 127
snmpauthprotocol	<String>	Read-write	SNMP v3 auth protocol for this profile.
ssh_port	<String>	Read-write	SSH port to connect to the device.
snmpprivprotocol	<String>	Read-write	SNMP v3 priv protocol for this profile.
host_password	<String>	Read-write	Host Password for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpversion	<String>	Read-write	SNMP version for this profile.
passphrase	<String>	Read-write	Passphrase with which private key is encrypted.
host_username	<String>	Read-write	Host User Name for this profile.Used for BLX form factor of ADC.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 127
snmpsecurityname	<String>	Read-write	SNMP v3 security name for this profile. Maximum length = 31
ssl_private_key	<String>	Read-write	SSL Private Key for key based authentication.
ssl_cert	<String>	Read-write	SSL Certificate for certificate based authentication.
http_port	<Integer>	Read-write	HTTP port to connect to the device.
snmpcommunity	<String>	Read-write	SNMP community for this profile. Maximum length = 31
https_port	<Integer>	Read-write	HTTPS port to connect to the device.
max_wait_time_reboot	<String>	Read-write	Max waiting time to reboot Citrix ADC.
snmpprivpassword	<String>	Read-write	SNMP v3 priv password for this profile. Minimum length = 8 Maximum length = 31

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cb_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix SD-WAN profiles.
snmpauthpassword	<String>	Read-write	SNMP v3 auth password for this profile. Minimum length = 8 Maximum length = 31
tenant_id	<String>	Read-only	Tenant Id of device profile.
is_default	<Boolean>	Read-only	True, if this profile is system generated and can not be deleted.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps_device_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ca_sps_device_profile": {

<b>"name":<String_value></b>,
```

"password":<String_value>,

"username":<String_value>,

"snmpsecuritylevel":<String_value>,

"svm_ns_comm":<String_value>,

"ssh_port":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"type":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"ns_profile_name":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

```
"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ca_sps_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,
```

```
"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps_device_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps_device_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ca_sps_device_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ca_sps_device_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ca_sps_device_profile **? pagesize=#no;pageno=#no** Use this query-parameter to get the ca_sps_device_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ca_sps_device_profile **?count=yes** Use this query-parameter to get the count of ca_sps_device_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ca_sps_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,
```

```
"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps_device_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ca_sps_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,
```

```
"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

ca_sps_downgrade

Configuration for Downgrade CA SPS Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ca_sps_ip_address_arr	<String[]>	Read-write	List of CA SPS IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- DOWNGRADE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

downgrade

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps_downgrade?action=downgrade;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ca_sps_downgrade": {

<b>"ca_sps_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ca_sps_downgrade":[{

"ca_sps_ip_address_arr":<String_value>}]}
```

ca_sps_init_config

Configuration for Initial Configuration Parameters of ca_sps resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
master_key_password	<String>	Read-write	Default Master Key Password.
fips_mode	<String>	Read-write	This parameter defines whether the Policy Server we register a trusted host registration with

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			is using FIPS mode or a non-FIPS mode.
enable_shared_secret_rollover	<Boolean>	Read-write	This parameter lets you change the shared secret that the CA SiteMinder Policy Server uses to encrypt communications to the SPS Agents..
policy_svr_ip	<String>	Read-write	Policy Server IP address.
host_config_obj_name	<String>	Read-write	Admin profile Password of Websense Protector instance.
parent_id	<String>	Read-write	.
agent_name	<String>	Read-write	Agent Name.
enable_fed_gateway	<Boolean>	Read-write	This parameter defines whether you want this CA SiteMinder SPS instance to act as a Federation Gateway..
id	<String>	Read-write	.
parent_name	<String>	Read-write	.
admin_reg_name	<String>	Read-write	Admin Reg Name.
agent_config_obj_name	<String>	Read-write	Agent Configuration Object Name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
trusted_host_name	<String>	Read-write	Trusted Host name which will be used to uniquely identify this CA SiteMinder SPS instance.
enable_webagent_result	<Boolean>	Read-write	This parameter defines whether you want this CA SiteMinder SPS instance to act as a Web Agent..
apache_svr_name	<String>	Read-write	APACHE Server Name.
admin_reg_password	<String>	Read-write	Admin Reg Password.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ca_sps_upgrade

Configuration for Upgrade CA SPS Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	CA SPS Image Name. Minimum length = 1 Maximum length = 128
ca_sps_ip_address_arr	<String[]>	Read-write	List of CA SPS IP Address. Minimum length = 1 Maximum length = 64
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- UPGRADE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

upgrade

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps_upgrade?action=upgrade;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ca_sps_upgrade": {

<b>"image_name":<String_value></b>,

<b>"ca_sps_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ca_sps_upgrade":[{

"act_id":<String_value>,

"image_name":<String_value>,

"ca_sps_ip_address_arr":<String_value>}]}
```

ca_sps_vpx_image

Configuration for XVA images of CASPS Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps_vpx_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps_vpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ca_sps_vpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ca_sps_vpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ca_sps_vpx_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the ca_sps_vpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ca_sps_vpx_image **?count=yes** Use this query-parameter to get the count of ca_sps_vpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ca_sps_vpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ca_sps_vpx_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ca_sps_vpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/ca_sps_vpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

Configuration-Audit

- config_advice
- config_advice_file
- config_advice_result
- ns_conf
- ns_conf_download_policy
- ns_conf_revision_history
- ns_conf_upgrade_history
- ns_config_audit_settings
- ns_config_diff
- ns_configtemplate
- ns_configtemplate_scheduler
- ns_filediff
- ns_filediff_settings

config_advice

Configuration for Config Advice resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_xendesktop	<Boolean>	Read-write	Enables search for Xen Desktop properties .
output_file	<String>	Read-write	File Path.
is_pci	<Boolean>	Read-write	Enables search for PCI properties.
config_advice_results	<config_advice_result[]>	Read-write	config advice result.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address	<String>	Read-write	IP Address.
id	<String>	Read-write	System generated key.
command	<String>	Read-write	Commands.

Operations

(click to see Properties)

- CUSTOM
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

custom

URL: https://<MGMT-IP>/nitro/v2/config/config_advice?action=custom;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"config_advice": {

"is_pci":<Boolean_value>,

"id":<String_value>,

"is_xendesktop":<Boolean_value>,

"ip_address":<String_value>,

"command":<String_value>,

"config_advice_results":[{

"modification_status_message":<String_value>,

"issue":<String_value>,

"corrective_command":<String_value>,
```

```
"category":<String_value>,

"id":<String_value>,

"advice":<String_value>,

"severity":<String_value>,

"modification_status":<Boolean_value>}],

"output_file":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"config_advice":[{

"is_xendesktop":<Boolean_value>,

"output_file":<String_value>,

"is_pci":<Boolean_value>,

"config_advice_results":[{

"parent_name":<String_value>,

"modification_status_message":<String_value>,

"issue":<String_value>,

"parent_id":<String_value>,

"modification_status":<Boolean_value>,

"corrective_command":<String_value>,
```

```
"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"advice":<String_value>}],

"ip_address":<String_value>,

"id":<String_value>,

"command":<String_value>}]}
```

config_advice_file

Configuration for Config Advice File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
is_main	<Boolean>	Read-write	File Location on Client for upload/ download.
file_size	<Integer>	Read-only	File Size.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/config_advice_file/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/config_advice_file **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/config_advice_file **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of config_advice_file resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/config_advice_file **? pagesize=#no;pageno=#no** Use this query-parameter to get the config_advice_file resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/config_advice_file **?count=yes** Use this query-parameter to get the count of config_advice_file resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"config_advice_file":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_main":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/config_advice_file/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"config_advice_file":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_main":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/config_advice_file **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

config_advice_result

Configuration for Config advice result resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
modification_status_message	<String>	Read-write	Command Deployment message.
issue	<String>	Read-write	Issue.
modification_status	<Boolean>	Read-write	Command Deployment Status.
corrective_command	<String>	Read-write	Command.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 256
id	<String>	Read-write	System generated key.
category	<String>	Read-write	Category.
severity	<String>	Read-write	Severity.
advice	<String>	Read-write	Advice.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ns_conf

Configuration for Configuration diff resource for Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String>	Read-write	List of Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key for all ns.conf files.
diff_reportname	<String>	Read-only	Name of the SavedVsRunning diff reportname.
hostname	<String>	Read-only	Host Name of the device.
diff_status	<String>	Read-only	Tells whether a diff exists between running and saved configuration on NS.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
display_name	<String>	Read-only	Display Name of the device.
last_updated_time	<Integer>	Read-only	Last Updated Time.
device_name	<String>	Read-only	Name of the device.
templatediff_reportname	<String>	Read-only	Name of the Template diff reports.
templatediff_status	<String>	Read-only	Tells whether a diff exists between templates associated to Citrix ADC and the running configuration.

Operations

(click to see Properties)

- INVENTORY
- DOWNLOAD_CONF
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

inventory

URL: https://<MGMT-IP>/nitro/v2/config/ns_conf?action=inventory;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_conf": {

"ns_ip_address":<String_value>,

"id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_conf":[{

"diff_reportname":<String_value>,

"hostname":<String_value>,

"diff_status":<String_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>,

"last_updated_time":<Integer_value>,

"device_name":<String_value>,

"templatediff_reportname":<String_value>,

"id":<String_value>,

"templatediff_status":<String_value>,

"last_updated_times":<String_value>}]}
```

download_conf

URL: https://<MGMT-IP>/nitro/v2/config/ns_conf?action=download_conf;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_conf": {

"ns_ip_address":<String_value>,

"id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_conf":[{

"diff_reportname":<String_value>,

"hostname":<String_value>,

"diff_status":<String_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>,

"last_updated_time":<Integer_value>,

"device_name":<String_value>,

"templatediff_reportname":<String_value>,

"id":<String_value>,

"templatediff_status":<String_value>,

"last_updated_times":<String_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_conf **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_conf **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_conf resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_conf **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_conf resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_conf **?count=yes** Use this query-parameter to get the count of ns_conf resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_conf":[{

"diff_reportname":<String_value>,

"hostname":<String_value>,
```

```
"diff_status":<String_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>,

"last_updated_time":<Integer_value>,

"device_name":<String_value>,

"templatediff_reportname":<String_value>,

"id":<String_value>,

"templatediff_status":<String_value>,

"last_updated_times":<String_value>}]}
```

ns_conf_download_policy

Configuration for Citrix ADC Configuration (ns.conf) Download Policy resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
interval_unit	<String>	Read-write	This property is deprecated, polling interval will be provided in minutes only;ltbr;gtFrequency unit (Minutes).
polling_interval	<Integer>	Read-write	Frequency of polling in minutes. Minimum value = 30 Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String[]>	Read-write	List of Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
act_id	<String>	Read-write	Activity ID that is used to track the polling status using resource poll_activity_status..

Operations

(click to see Properties)

- DO_POLL
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

do_poll

URL: https://<MGMT-IP>/nitro/v2/config/ns_conf_download_policy?action=do_poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_conf_download_policy": {

<b>"polling_interval":<Integer_value></b>,

"ns_ip_address":<String_value[]>,

"act_id":<String_value>,

"interval_unit":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_conf_download_policy":[{

"interval_unit":<String_value>,

"polling_interval":<Integer_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_conf_download_policy **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_conf_download_policy":[{

"interval_unit":<String_value>,

"polling_interval":<Integer_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/ns_conf_download_policy/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_conf_download_policy":{

<b>"polling_interval":<Integer_value></b>,

"ns_ip_address":<String_value[]>,

"act_id":<String_value>,

"interval_unit":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_conf_download_policy":[{

"interval_unit":<String_value>,

"polling_interval":<Integer_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

ns_conf_revision_history

Configuration for ns.conf file revision history on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
ns_conf_revisionfile	<String>	Read-only	ns.conf revision history file.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_conf_revision_history **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_conf_revision_history **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_conf_revision_history resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_conf_revision_history **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_conf_revision_history resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_conf_revision_history **?count=yes** Use this query-parameter to get the count of ns_conf_revision_history resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_conf_revision_history":[{

"ns_conf_revisionfile":<String_value>,

"ns_ip_address":<String_value>}]}
```

ns_conf_upgrade_history

Configuration for ns.conf file upgrade history on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
ns_conf_upgradefile	<String>	Read-only	ns.conf upgrade history file.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_conf_upgrade_history **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_conf_upgrade_history **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_conf_upgrade_history resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_conf_upgrade_history **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_conf_upgrade_history resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_conf_upgrade_history **?count=yes** Use this query-parameter to get the count of ns_conf_upgrade_history resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_conf_upgrade_history":[{

"ns_conf_upgradefile":<String_value>,

"ns_ip_address":<String_value>}]}
```

ns_config_audit_settings

Configuration for Config Audit Setting resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
audit_on_trap_receive	<Boolean>	Read-write	Run config audit for an ADC instance after config save trap is received.
is_template_diff_bidirectional	<Boolean>	Read-write	Indicates whether template Vs running diff is bi-directional.
id	<String>	Read-write	Id is system generated key..
time_after_receive	<Integer>	Read-write	Time (in minutes) after which Config Audit is run for a particular IP after config save trap is received.

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_config_audit_settings **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_config_audit_settings **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_config_audit_settings resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_config_audit_settings **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_config_audit_settings resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_config_audit_settings **?count=yes** Use this query-parameter to get the count of ns_config_audit_settings resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_config_audit_settings":[{

"audit_on_trap_receive":<Boolean_value>,

"is_template_diff_bidirectional":<Boolean_value>,

"id":<String_value>,

"time_after_receive":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_config_audit_settings/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_config_audit_settings":[{

"audit_on_trap_receive":<Boolean_value>,

"is_template_diff_bidirectional":<Boolean_value>,

"id":<String_value>,

"time_after_receive":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/ns_config_audit_settings/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_config_audit_settings":{

<b>"id":<String_value></b>,

"is_template_diff_bidirectional":<Boolean_value>,

"time_after_receive":<Integer_value>,

"audit_on_trap_receive":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_config_audit_settings":[{

"audit_on_trap_receive":<Boolean_value>,

"is_template_diff_bidirectional":<Boolean_value>,

"id":<String_value>,

"time_after_receive":<Integer_value>}]}
```

ns_config_diff

Configuration for Citrix ADC Configuration Diff resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
diff_file	<String>	Read-write	Name of the file containing diff between source and target file generated by a schedule diff operation..
source_file	<String>	Read-write	Citrix ADC configuration source file.
ignoresysuserpassword	<Boolean>	Read-write	When set to true, ignores password difference in diff command output.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
target_file	<String>	Read-write	Citrix ADC configuration target file.
corrective_command	<String>	Read-write	Corrective command.
diff_data	<String>	Read-only	Diff data in XML format..
params_targetvalue	<String>	Read-only	Params target value seen in the diff output.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
params_value	<String>	Read-only	Params value seen in the diff output.
diff_response	<String>	Read-only	Diff between source and target file.
source_command	<String>	Read-only	Diff command ouput present in source.
target_command	<String>	Read-only	Diff command ouput present in target.
params_name	<String>	Read-only	Params name seen in the diff output.

Operations

(click to see Properties)

- DIFF_TABLE_REPORT
- DIFF_TABLE
- DO_DIFF
- GEN_BATCH

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

diff_table_report

URL: https://<MGMT-IP>/nitro/v2/config/ns_config_diff?action=diff_table_report;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_config_diff": {

"ns_ip_address":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"target_file":<String_value>,
```

```
"corrective_command":<String_value>,

"source_file":<String_value>,

"diff_file":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_config_diff":[{

"diff_data":<String_value>,

"params_targetvalue":<String_value>,

"diff_file":<String_value>,

"source_file":<String_value>,

"params_value":<String_value>,

"diff_response":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"source_command":<String_value>,

"ns_ip_address":<String_value>,

"target_file":<String_value>,

"corrective_command":<String_value>,

"target_command":<String_value>,

"params_name":<String_value>}]}
```

diff_table

URL: https://<MGMT-IP>/nitro/v2/config/ns_config_diff?action=diff_table;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_config_diff": {

"ns_ip_address":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"target_file":<String_value>,

"corrective_command":<String_value>,

"source_file":<String_value>,

"diff_file":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_config_diff":[{
```

```
"diff_data":<String_value>,

"params_targetvalue":<String_value>,

"diff_file":<String_value>,

"source_file":<String_value>,

"params_value":<String_value>,

"diff_response":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"source_command":<String_value>,

"ns_ip_address":<String_value>,

"target_file":<String_value>,

"corrective_command":<String_value>,

"target_command":<String_value>,

"params_name":<String_value>}]}
```

do_diff

URL: https://<MGMT-IP>/nitro/v2/config/ns_config_diff?action=do_diff;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_config_diff": {

"ns_ip_address":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"target_file":<String_value>,

"corrective_command":<String_value>,

"source_file":<String_value>,

"diff_file":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_config_diff":[{

"diff_data":<String_value>,

"params_targetvalue":<String_value>,

"diff_file":<String_value>,

"source_file":<String_value>,

"params_value":<String_value>,

"diff_response":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"source_command":<String_value>,

"ns_ip_address":<String_value>,
```

```
"target_file":<String_value>,

"corrective_command":<String_value>,

"target_command":<String_value>,

"params_name":<String_value>}]}
```

gen_batch

URL: https://<MGMT-IP>/nitro/v2/config/ns_config_diff?action=gen_batch;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_config_diff": {

"ns_ip_address":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"target_file":<String_value>,

"corrective_command":<String_value>,

"source_file":<String_value>,

"diff_file":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_config_diff":[{

"diff_data":<String_value>,

"params_targetvalue":<String_value>,

"diff_file":<String_value>,

"source_file":<String_value>,

"params_value":<String_value>,

"diff_response":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"source_command":<String_value>,

"ns_ip_address":<String_value>,

"target_file":<String_value>,

"corrective_command":<String_value>,

"target_command":<String_value>,

"params_name":<String_value>}]}
```

ns_configtemplate

Configuration for Citrix ADC configuration template resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
to_be_exported	<Boolean>	Read-write	Setting for enabling/ disabling config audit diff report.
variable_vals	<variable_values[]>	Read-write	Values of Variables used in commands of the audit template.
commands	<ns_command[]>	Read-write	Citrix ADC Configuration Commands.
variables	<config_variable[]>	Read-write	Array of variables used in commands of the configuration template.
slack_profile	<String>	Read-write	Slack profile name to send slack notification.. Minimum length = 1 Maximum length = 255
mail_export	<Boolean>	Read-write	Setting for enabling/ disabling config audit diff report for mail export.
template_file	<String>	Read-write	CSV file for importing variables.
send_report_only_when_diff	<Boolean>	Read-write	Send notification report only when there is a diff.
id	<String>	Read-write	Id is system generated key for all the system users.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
mail_profile	<String>	Read-write	Mail profile name to send notification mail.. Minimum length = 1 Maximum length = 255
slack_export	<Boolean>	Read-write	Setting for enabling/ disabling config audit diff report for slack export.
isDefaultSchedule	<Boolean>	Read-write	Scheduled specification for template or default polling interval.
name	<String>	Read-write	Name of the configuration template. Minimum length = 1 Maximum length = 128
description	<String>	Read-write	Template Description. Minimum length = 1 Maximum length = 128
ignoresysuserpassword	<Boolean>	Read-write	When set to true, ignores password difference in diff command output.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
device_groups_db	<String>	Read-write	Device Groups List Comma separated to store in DB.
scheduleInfo	<ns_configtemplate_scheduler>	Read-write	Scheduling information for audit template.
preview_commands	<String[]>	Read-write	Preview of list of commands.
device_groups	<String[]>	Read-write	Device Group Array for which audit template is added.
devices	<String[]>	Read-write	List of Citrix ADC IP Address.
scheduleDetail	<String>	Read-only	Schedule Detail.
last_updated_time	<Integer>	Read-only	Last Updated Time.
tenant_id	<String>	Read-only	Tenant Id.
template_create_user	<String>	Read-only	User who created the template.

Operations

(click to see Properties)

- IMPORT_VARIABLES_FILE
- ADD
- PREVIEW
- DELETE
- GET (ALL)
- GET
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

import_variables_file

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate?action=import_variables_file;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_configtemplate": {

"name":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"mail_export":<Boolean_value>,

"id":<String_value>,

"devices":<String_value[]>,

"send_report_only_when_diff":<Boolean_value>,

"slack_export":<Boolean_value>,

"device_groups":<String_value[]>,

"preview_commands":<String_value[]>,

"description":<String_value>,

"scheduleInfo":<ns_configtemplate_scheduler_value>,

"variables":[{

<b>"name":<String_value></b>,

"parent_id":<String_value>,

"required":<Boolean_value>,

"default_value":<String_value>,

"id":<String_value>,
```

```
"values_enum":<String_value[]>,

"values_enum_db":<String_value>,

"parent_name":<String_value>,

"description":<String_value>,

"type":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>}],

"isDefaultSchedule":<Boolean_value>,

"slack_profile":<String_value>,

"template_file":<String_value>,

"mail_profile":<String_value>,

"device_groups_db":<String_value>,

"to_be_exported":<Boolean_value>,

"variable_vals":[{

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"values_enum_db":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"description":<String_value>,
```

```
"type":<String_value>,

"valuex":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"device":<String_value>,

"device_group":<String_value>,

"valuex":<String_value>,

"parent_id":<String_value>}],

"display_name":<String_value>}],

"commands":[{

<b>"id":<String_value></b>,

"parent_id":<String_value>,

"parent_name":<String_value>,

"command":<String_value>}]}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_configtemplate":[{

"to_be_exported":<Boolean_value>,

"variable_vals":[{

"device_values_mapping":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"valuex":<String_value>,

"device_group":<String_value>,

"device":<String_value>,

"parent_id":<String_value>}],

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,
```

```
"type":<String_value>,

"values_enum_db":<String_value>}],

"devicelistCombined":<String_value>,

"commands":[{

"index":<Integer_value>,

"parent_name":<String_value>,

"id":<String_value>,

"command":<String_value>,

"parent_id":<String_value>}],

"variables":[{

"parent_name":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"parent_id":<String_value>,

"required":<Boolean_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>,

"values_enum":<String_value>}],
```

```
"slack_profile":<String_value>,

"scheduleDetail":<String_value>,

"last_updated_time":<Integer_value>,

"tenant_id":<String_value>,

"mail_export":<Boolean_value>,

"template_file":<String_value>,

"send_report_only_when_diff":<Boolean_value>,

"id":<String_value>,

"mail_profile":<String_value>,

"slack_export":<Boolean_value>,

"isDefaultSchedule":<Boolean_value>,

"name":<String_value>,

"devicelist":<String_value>,

"description":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"template_create_user":<String_value>,

"device_groups_db":<String_value>,

"scheduleInfo":<ns_configtemplate_scheduler_value>,

"preview_commands":<String_value>,

"device_groups":<String_value>,

"devices":<String_value>}]}
```

add

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate?onerror=<String_value> **HTTPS**
Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_configtemplate": {

<b>"commands":[{

<b>"id":<String_value></b>,

"parent_id":<String_value>,

"parent_name":<String_value>,

"command":<String_value>}]</b>,

<b>"name":<String_value></b>,

<b>"devices":<String_value[]></b>,

"ignoresysuserpassword":<Boolean_value>,

"mail_export":<Boolean_value>,

"id":<String_value>,

"send_report_only_when_diff":<Boolean_value>,

"slack_export":<Boolean_value>,
```

```
"device_groups":<String_value[]>,

"preview_commands":<String_value[]>,

"description":<String_value>,

"scheduleInfo":<ns_configtemplate_scheduler_value>,

"variables":[{

<b>"name":<String_value></b>,

"parent_id":<String_value>,

"required":<Boolean_value>,

"default_value":<String_value>,

"id":<String_value>,

"values_enum":<String_value[]>,

"values_enum_db":<String_value>,

"parent_name":<String_value>,

"description":<String_value>,

"type":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>}],

"isDefaultSchedule":<Boolean_value>,

"slack_profile":<String_value>,

"template_file":<String_value>,

"mail_profile":<String_value>,

"device_groups_db":<String_value>,
```

```
"to_be_exported":<Boolean_value>,

"variable_vals":[{

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"values_enum_db":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"description":<String_value>,

"type":<String_value>,

"valuex":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"device":<String_value>,

"device_group":<String_value>,

"valuex":<String_value>,

"parent_id":<String_value>}],

"display_name":<String_value>}]}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_configtemplate":[{

"to_be_exported":<Boolean_value>,

"variable_vals":[{

"device_values_mapping":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"valuex":<String_value>,

"device_group":<String_value>,

"device":<String_value>,
```

```
"parent_id":<String_value>}],

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>}],

"devicelistCombined":<String_value>,

"commands":[{

"index":<Integer_value>,

"parent_name":<String_value>,

"id":<String_value>,

"command":<String_value>,

"parent_id":<String_value>}],

"variables":[{

"parent_name":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"parent_id":<String_value>,

"required":<Boolean_value>,

"default_value":<String_value>,
```

```
"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>,

"values_enum":<String_value>}],

"slack_profile":<String_value>,

"scheduleDetail":<String_value>,

"last_updated_time":<Integer_value>,

"tenant_id":<String_value>,

"mail_export":<Boolean_value>,

"template_file":<String_value>,

"send_report_only_when_diff":<Boolean_value>,

"id":<String_value>,

"mail_profile":<String_value>,

"slack_export":<Boolean_value>,

"isDefaultSchedule":<Boolean_value>,

"name":<String_value>,

"devicelist":<String_value>,

"description":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"template_create_user":<String_value>,

"device_groups_db":<String_value>,

"scheduleInfo":<ns_configtemplate_scheduler_value>,
```

```
"preview_commands":<String_value>,

"device_groups":<String_value>,

"devices":<String_value>}]}
```

preview

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate?action=preview;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_configtemplate": {

"name":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"mail_export":<Boolean_value>,

"id":<String_value>,

"devices":<String_value[]>,

"send_report_only_when_diff":<Boolean_value>,

"slack_export":<Boolean_value>,

"device_groups":<String_value[]>,

"preview_commands":<String_value[]>,
```

```
"description":<String_value>,

"scheduleInfo":<ns_configtemplate_scheduler_value>,

"variables":[{

<b>"name":<String_value></b>,

"parent_id":<String_value>,

"required":<Boolean_value>,

"default_value":<String_value>,

"id":<String_value>,

"values_enum":<String_value[]>,

"values_enum_db":<String_value>,

"parent_name":<String_value>,

"description":<String_value>,

"type":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>}],

"isDefaultSchedule":<Boolean_value>,

"slack_profile":<String_value>,

"template_file":<String_value>,

"mail_profile":<String_value>,

"device_groups_db":<String_value>,

"to_be_exported":<Boolean_value>,

"variable_vals":[{
```

```
"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"values_enum_db":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"description":<String_value>,

"type":<String_value>,

"valuex":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"device":<String_value>,

"device_group":<String_value>,

"valuex":<String_value>,

"parent_id":<String_value>}],

"display_name":<String_value>}],

"commands":[{

<b>"id":<String_value></b>,

"parent_id":<String_value>,
```

```
"parent_name":<String_value>,

"command":<String_value>}]}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_configtemplate":[{

"to_be_exported":<Boolean_value>,

"variable_vals":[{

"device_values_mapping":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"valuex":<String_value>,
```

```
"device_group":<String_value>,

"device":<String_value>,

"parent_id":<String_value>}],

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>}],

"devicelistCombined":<String_value>,

"commands":[{

"index":<Integer_value>,

"parent_name":<String_value>,

"id":<String_value>,

"command":<String_value>,

"parent_id":<String_value>}],

"variables":[{

"parent_name":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"parent_id":<String_value>,
```

```
"required":<Boolean_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>,

"values_enum":<String_value>}],

"slack_profile":<String_value>,

"scheduleDetail":<String_value>,

"last_updated_time":<Integer_value>,

"tenant_id":<String_value>,

"mail_export":<Boolean_value>,

"template_file":<String_value>,

"send_report_only_when_diff":<Boolean_value>,

"id":<String_value>,

"mail_profile":<String_value>,

"slack_export":<Boolean_value>,

"isDefaultSchedule":<Boolean_value>,

"name":<String_value>,

"devicelist":<String_value>,

"description":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"template_create_user":<String_value>,
```

```
"device_groups_db":<String_value>,

"scheduleInfo":<ns_configtemplate_scheduler_value>,

"preview_commands":<String_value>,

"device_groups":<String_value>,

"devices":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_configtemplate **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_configtemplate resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_configtemplate **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_configtemplate resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_configtemplate **?count=yes** Use this query-parameter to get the count of ns_configtemplate resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_configtemplate":[{

"to_be_exported":<Boolean_value>,

"variable_vals":[{

"device_values_mapping":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,
```

```
"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"valuex":<String_value>,

"device_group":<String_value>,

"device":<String_value>,

"parent_id":<String_value>}],

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>}],

"devicelistCombined":<String_value>,

"commands":[{

"index":<Integer_value>,

"parent_name":<String_value>,

"id":<String_value>,

"command":<String_value>,
```

```
"parent_id":<String_value>}],

"variables":[{

"parent_name":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"parent_id":<String_value>,

"required":<Boolean_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>,

"values_enum":<String_value>}],

"slack_profile":<String_value>,

"scheduleDetail":<String_value>,

"last_updated_time":<Integer_value>,

"tenant_id":<String_value>,

"mail_export":<Boolean_value>,

"template_file":<String_value>,

"send_report_only_when_diff":<Boolean_value>,

"id":<String_value>,
```

```
"mail_profile":<String_value>,

"slack_export":<Boolean_value>,

"isDefaultSchedule":<Boolean_value>,

"name":<String_value>,

"devicelist":<String_value>,

"description":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"template_create_user":<String_value>,

"device_groups_db":<String_value>,

"scheduleInfo":<ns_configtemplate_scheduler_value>,

"preview_commands":<String_value>,

"device_groups":<String_value>,

"devices":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_configtemplate":[{

"to_be_exported":<Boolean_value>,

"variable_vals":[{

"device_values_mapping":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"valuex":<String_value>,

"device_group":<String_value>,
```

```
"device":<String_value>,

"parent_id":<String_value>}],

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>}],

"devicelistCombined":<String_value>,

"commands":[{

"index":<Integer_value>,

"parent_name":<String_value>,

"id":<String_value>,

"command":<String_value>,

"parent_id":<String_value>}],

"variables":[{

"parent_name":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"parent_id":<String_value>,

"required":<Boolean_value>,
```

```
"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>,

"values_enum":<String_value>}],

"slack_profile":<String_value>,

"scheduleDetail":<String_value>,

"last_updated_time":<Integer_value>,

"tenant_id":<String_value>,

"mail_export":<Boolean_value>,

"template_file":<String_value>,

"send_report_only_when_diff":<Boolean_value>,

"id":<String_value>,

"mail_profile":<String_value>,

"slack_export":<Boolean_value>,

"isDefaultSchedule":<Boolean_value>,

"name":<String_value>,

"devicelist":<String_value>,

"description":<String_value>,

"ignoresysuserpassword":<Boolean_value>,

"template_create_user":<String_value>,

"device_groups_db":<String_value>,
```

```
"scheduleInfo":<ns_configtemplate_scheduler_value>,

"preview_commands":<String_value>,

"device_groups":<String_value>,

"devices":<String_value>}]}
```

modify

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_configtemplate":{

<b>"commands":[{

<b>"id":<String_value></b>,

"parent_id":<String_value>,

"parent_name":<String_value>,

"command":<String_value>}]</b>,

<b>"id":<String_value></b>,

<b>"devices":<String_value[]></b>,

"name":<String_value>,
```

```
"ignoresysuserpassword":<Boolean_value>,

"mail_export":<Boolean_value>,

"send_report_only_when_diff":<Boolean_value>,

"slack_export":<Boolean_value>,

"device_groups":<String_value[]>,

"preview_commands":<String_value[]>,

"description":<String_value>,

"scheduleInfo":<ns_configtemplate_scheduler_value>,

"variables":[{

<b>"name":<String_value></b>,

"parent_id":<String_value>,

"required":<Boolean_value>,

"default_value":<String_value>,

"id":<String_value>,

"values_enum":<String_value[]>,

"values_enum_db":<String_value>,

"parent_name":<String_value>,

"description":<String_value>,

"type":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>}],

"isDefaultSchedule":<Boolean_value>,
```

```
"slack_profile":<String_value>,

"template_file":<String_value>,

"mail_profile":<String_value>,

"device_groups_db":<String_value>,

"to_be_exported":<Boolean_value>,

"variable_vals":[{

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"values_enum_db":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"description":<String_value>,

"type":<String_value>,

"valuex":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"device":<String_value>,

"device_group":<String_value>,
```

```
"valuex":<String_value>,

"parent_id":<String_value>}],

"display_name":<String_value>}]}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_configtemplate":[{

"to_be_exported":<Boolean_value>,

"variable_vals":[{

"device_values_mapping":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,
```

```
"valuex":<String_value>,

"device_group":<String_value>,

"device":<String_value>,

"parent_id":<String_value>}],

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>}],

"devicelistCombined":<String_value>,

"commands":[{

"index":<Integer_value>,

"parent_name":<String_value>,

"id":<String_value>,

"command":<String_value>,

"parent_id":<String_value>}],

"variables":[{

"parent_name":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,
```

```
"parent_id":<String_value>,

"required":<Boolean_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>,

"values_enum":<String_value>}],

"slack_profile":<String_value>,

"scheduleDetail":<String_value>,

"last_updated_time":<Integer_value>,

"tenant_id":<String_value>,

"mail_export":<Boolean_value>,

"template_file":<String_value>,

"send_report_only_when_diff":<Boolean_value>,

"id":<String_value>,

"mail_profile":<String_value>,

"slack_export":<Boolean_value>,

"isDefaultSchedule":<Boolean_value>,

"name":<String_value>,

"devicelist":<String_value>,

"description":<String_value>,

"ignoresysuserpassword":<Boolean_value>,
```

```
"template_create_user":<String_value>,

"device_groups_db":<String_value>,

"scheduleInfo":<ns_configtemplate_scheduler_value>,

"preview_commands":<String_value>,

"device_groups":<String_value>,

"devices":<String_value>}]}
```

ns_configtemplate_scheduler

Configuration for config audit template scheduler resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
scheduleTimesEpoch	<String>	Read-write	Epoch times for scheduling the job.
daily_time	<String>	Read-write	Daily time details of scheduler.
recurrenceOptions	<String>	Read-write	Comma separated recurrence options of job that is scheduled.
id	<String>	Read-write	Id is system generated key.
recurrence_type	<String>	Read-write	Daily, Week or Monthly option. Maximum length = 255
tz_offset	<Integer>	Read-write	Time zone offset..

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate_scheduler?onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_configtemplate_scheduler": {

"daily_time":<String_value>,

"recurrence_type":<String_value>,

"id":<String_value>,

"scheduleTimesEpoch":<String_value>,

"tz_offset":<Integer_value>,

"recurrenceOptions":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_configtemplate_scheduler":[{

"scheduleTimesEpoch":<String_value>,
```

```
"parent_name":<String_value>,

"daily_time":<String_value>,

"recurrenceOptions":<String_value>,

"id":<String_value>,

"recurrence_type":<String_value>,

"tz_offset":<Integer_value>,

"parent_id":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate_scheduler/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate_scheduler **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_configtemplate_scheduler **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_configtemplate_scheduler resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_configtemplate_scheduler **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_configtemplate_scheduler resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_configtemplate_scheduler **?count=yes** Use this query-parameter to get the count of ns_configtemplate_scheduler resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_configtemplate_scheduler":[{

"scheduleTimesEpoch":<String_value>,

"parent_name":<String_value>,

"daily_time":<String_value>,

"recurrenceOptions":<String_value>,
```

```
"id":<String_value>,

"recurrence_type":<String_value>,

"tz_offset":<Integer_value>,

"parent_id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate_scheduler/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_configtemplate_scheduler":[{

"scheduleTimesEpoch":<String_value>,

"parent_name":<String_value>,

"daily_time":<String_value>,
```

```
"recurrenceOptions":<String_value>,

"id":<String_value>,

"recurrence_type":<String_value>,

"tz_offset":<Integer_value>,

"parent_id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/ns_configtemplate_scheduler/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_configtemplate_scheduler":{

<b>"id":<String_value></b>,

"daily_time":<String_value>,

"recurrence_type":<String_value>,

"scheduleTimesEpoch":<String_value>,

"tz_offset":<Integer_value>,

"recurrenceOptions":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_configtemplate_scheduler":[{

"scheduleTimesEpoch":<String_value>,

"parent_name":<String_value>,

"daily_time":<String_value>,

"recurrenceOptions":<String_value>,

"id":<String_value>,

"recurrence_type":<String_value>,

"tz_offset":<Integer_value>,

"parent_id":<String_value>}]}
```

ns_filediff

Configuration for monitoring file list diff present in nsconfig folder of ADC instances resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
previous_filelist_details	<String>	Read-write	Comma separated list of files under nsconfig folder found in previous poll.
diff_summary	<String>	Read-write	diff summary.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
display_name	<String>	Read-write	display name .
latest_filelist_details	<String>	Read-write	Comma separated list of files under nsconfig folder found in latest poll.
host_name	<String>	Read-write	host name.
id	<String>	Read-write	Id is system generated key.
diff_exists	<String>	Read-write	Diff exists. Possible values No Diff,NA,Diff Exists.
filediff_na_count	<Integer>	Read-write	Total file diff not applicable count .
file_name	<String>	Read-write	file name.
last_modified	<Double>	Read-write	last modified timestamp of the file.
filediff_count	<Integer>	Read-write	Total file diff count .
diff_status	<String>	Read-write	diff status of the file. Possible values: added,modified,deleted.
total_count	<Integer>	Read-write	Total count available .
latest_polled_timestamp	<Double>	Read-only	Latest polled timestamp.
previous_polled_timestamp	<Double>	Read-only	Previous polled timestamp.
ip_address	<String>	Read-only	IP Address of ADC device.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_filediff **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_filediff **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_filediff resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_filediff **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_filediff resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_filediff **?count=yes** Use this query-parameter to get the count of ns_filediff resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_filediff":[{

"latest_polled_timestamp":<Double_value>,

"previous_filelist_details":<String_value>,

"diff_summary":<String_value>,

"display_name":<String_value>,

"latest_filelist_details":<String_value>,
```

```
"previous_polled_timestamp":<Double_value>,

"host_name":<String_value>,

"id":<String_value>,

"ip_address":<String_value>,

"diff_exists":<String_value>,

"filediff_na_count":<Integer_value>,

"file_name":<String_value>,

"last_modified":<Double_value>,

"filediff_count":<Integer_value>,

"diff_status":<String_value>,

"total_count":<Integer_value>}]}
```

ns_filediff_settings

Configuration for Mail settings for Config File Diff feature resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
slack_profile	<String>	Read-write	slack profile .
mail_profile	<String>	Read-write	mail profile .

Operations

(click to see Properties)

- GET (ALL)

- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_filediff_settings **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_filediff_settings **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of

ns_filediff_settings resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_filediff_settings **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_filediff_settings resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_filediff_settings **?count=yes** Use this query-parameter to get the count of ns_filediff_settings resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_filediff_settings":[{

"slack_profile":<String_value>,

"mail_profile":<String_value>}]}
```

modify

URL: https://<MGMT-IP>/nitro/v2/config/ns_filediff_settings/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_filediff_settings":{

"slack_profile":<String_value>,

"mail_profile":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_filediff_settings":[{

"slack_profile":<String_value>,

"mail_profile":<String_value>}]}
```

Configuration-Jobs

- config_variable
- device_values_map
- variable_values

config_variable

Configuration for Configuration Variables for use in configuration template resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_name	<String>	Read-write	Parent name of config variable.
name	<String>	Read-write	Variable name. Minimum length = 1 Maximum length = 128
valuex	<String>	Read-write	Secure Value of the Variable to be used for all device ips/ groups.
display_name	<String>	Read-write	Display name of configuration variable. Minimum length = 1 Maximum length = 1024
description	<String>	Read-write	Description of configuration variable. Minimum length = 1 Maximum length = 1024
parent_id	<String>	Read-write	Parent Id of config variable.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
required	<Boolean>	Read-write	If it is required argument for command then true else false..
default_value	<String>	Read-write	Default Value of configuration variable.
id	<String>	Read-write	Id is system generated key for all config variable.
type	<String>	Read-write	Type of Variable Text Field, Numeric Text Field, Choice, File, IP Addrss Field, Password Field. Minimum length = 1 Maximum length = 64
values_enum_db	<String>	Read-write	Comma separated list of possible values of variable.
values_enum	<String[]>	Read-write	Possible Values values for choice type variable.

Operations

(click to see Properties)

- ADD
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/config_variable?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"config_variable": {

<b>"name":<String_value></b>,

"parent_id":<String_value>,

"required":<Boolean_value>,

"default_value":<String_value>,

"id":<String_value>,

"values_enum":<String_value[]>,

"values_enum_db":<String_value>,

"parent_name":<String_value>,

"description":<String_value>,

"type":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"config_variable":[{

"parent_name":<String_value>,
```

```
"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"parent_id":<String_value>,

"required":<Boolean_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>,

"values_enum":<String_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/config_variable **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/config_variable **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of config_variable resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/config_variable **? pagesize=#no;pageno=#no** Use this query-parameter to get the config_variable resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/config_variable **?count=yes** Use this query-parameter to get the count of config_variable resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"config_variable":[{

"parent_name":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"parent_id":<String_value>,

"required":<Boolean_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>,

"values_enum":<String_value>}]}
```

device_values_map

Configuration for Map of Device and value to be used for it resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_name	<String>	Read-write	Parent Name.
value	<String>	Read-write	Value of the Variable to be used for device ip/group.
id	<String>	Read-write	Id is system generated key for all the device value mapping.
valuex	<String>	Read-write	Secure Value of the Variable to be used for device ip/group.
device_group	<String>	Read-write	Device Group for which value is to be defined.
device	<String>	Read-write	IP Address for this device.
parent_id	<String>	Read-write	Parent ID.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

variable_values

Configuration for Variable Values required for config template execution resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_name	<String>	Read-write	.
value	<String>	Read-write	Value of the Variable to be used for all device ips/groups.
name	<String>	Read-write	Variable name. Minimum length = 1 Maximum length = 128
valuex	<String>	Read-write	Secure Value of the Variable to be used for all device ips/ groups.
display_name	<String>	Read-write	Display name of configuration variable. Minimum length = 1 Maximum length = 1024
description	<String>	Read-write	Description of configuration variable. Minimum length = 1 Maximum length = 1024
device_values	<device_values_map []>	Read-write	Values of variables used in commands of the configuration template for individual devices.
parent_id	<String>	Read-write	.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
default_value	<String>	Read-write	Default Value of configuration variable.
id	<String>	Read-write	.
type	<String>	Read-write	Type of Variable. Minimum length = 1 Maximum length = 64
values_enum_db	<String>	Read-write	Comma separated list of possible values of variable.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/variable_values **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/variable_values **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of variable_values resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/variable_values **? pagesize=#no;pageno=#no** Use this query-parameter to get the variable_values resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/variable_values **?count=yes** Use this query-parameter to get the count of variable_values resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"variable_values":[{

"device_values_mapping":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"valuex":<String_value>,

"device_group":<String_value>,

"device":<String_value>,

"parent_id":<String_value>}],
```

```
"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>}]}
```

Device-SSL-Certificates

- ns_ssl_cert
- ns_ssl_certkey
- ns_ssl_certkey_policy
- ns_ssl_certlink
- ns_ssl_csr
- ns_ssl_files
- ns_ssl_key
- ssl_cert_threshold

ns_ssl_cert

Configuration for Citrix ADC SSL certificate File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 256
tenant_name	<String>	Read-only	Tenant Name.
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_cert/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_cert **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_ssl_cert **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_ssl_cert resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_ssl_cert **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_ssl_cert resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_ssl_cert **?count=yes** Use this query-parameter to get the count of ns_ssl_cert resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_cert":[{

"tenant_name":<String_value>,

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,
```

```
"file_last_modified":<String_value>,

"is_dir":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_cert/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_cert":[{

"tenant_name":<String_value>,

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,
```

```
"file_last_modified":<String_value>,

"is_dir":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/ns_ssl_cert **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

ns_ssl_certkey

Configuration for SSL certificate on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
no_domain_check	<Boolean>	Read-write	Specify this option to override the check for matching domain names during certificate update operation.
ns_ip_address	<String>	Read-write	List of Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
certkeypair_name	<String>	Read-write	Cert Key Pair Name. Minimum length = 1 Maximum length = 128
certificate_dn	<String>	Read-write	For certificates created by Venafi,this property indicates the certificate distinguished name..
ssl_key	<String>	Read-write	Key. Minimum length = 1 Maximum length = 128
id	<String>	Read-write	Id is system generated key for all ssl cert-keys entries. For download operation "id" must be provided in the format ;ltadc_ip_addr

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			ess;gt_;ltcertkeypair_name;gt.tgz.
key_data	<String>	Read-write	Key Data. Maximum length = 16384
managed_by	<String>	Read-write	Indicates if the certificate is managed by a third party certificate management provider..
ssl_certificate	<String>	Read-write	Certificate. Minimum length = 1 Maximum length = 128
certificate_data	<String>	Read-write	Certificate Data. Maximum length = 16384
file_location_path	<String>	Read-write	File Location on Client for download. Minimum length = 1
cert_format	<String>	Read-write	Certificate Format. Maximum length = 64
certchainbinding	<String[]>	Read-write	Certificate Chain binding..
app_name	<String>	Read-write	Application name.Used for tracking task logs created for a specific application.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
password	<String>	Read-write	The pass-phrase that was used to encrypt the private-key.. Maximum length = 128
source_certificate	<String>	Read-write	CertKeyPair Name of the certificate that needs to installed in another instance.
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
certkeypair_arr	<String[]>	Read-write	List of certkeypair_name.Used in cert binding operation..
source_ipaddress	<String>	Read-write	Citrix ADC IP of the certificate that needs to installed in another instance.
save_config	<Boolean>	Read-write	true, if save config is required.
ns_ip_address_arr	<String[]>	Read-write	List of Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
entity_name	<String>	Read-write	entity name.
snicert	<Boolean>	Read-write	Indicates server certificate for SNI.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Used in cert binding operation..
serial_number	<String>	Read-only	Serial Number.
signature_algorithm	<String>	Read-only	Signature Algorithm.
valid_from	<String>	Read-only	Valid From.
status	<String>	Read-only	Tells whether the certificate is still valid or not.
hostname	<String>	Read-only	Host Name of the device.
issuer	<String>	Read-only	Issuer.
public_key_size	<Integer>	Read-only	Public Key Size.
device_name	<String>	Read-only	Name of the device.
valid_to	<String>	Read-only	Valid To.
subject	<String>	Read-only	Subject.
version	<Integer>	Read-only	Version.
public_key_algorithm	<String>	Read-only	Public Key Algorithm.
days_to_expiry	<Integer>	Read-only	Days before SSL certificate expires.
poll_time	<Integer>	Read-only	Last Polling Time.
display_name	<String>	Read-only	Display Name of the device.
no_of_bound_entities	<Integer>	Read-only	no_of_bound_entities.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
partition_name	<String>	Read-only	Name of Admin Partition. Blank means Default Partition.
subjaltname	<String>	Read-only	Comma separated Subject Alternative Names.
domain	<String>	Read-only	Domain name for certificate issuer.
datacenter_id	<String>	Read-only	Datacenter Id .
activity_id	<String>	Read-only	Activity Id used to track the progress of certificate operations..
csr	<String>	Read-only	Certificate Signing Request.

Operations

(click to see Properties)

- INVENTORY
- LIST_ENTITY_CERT
- BIND_CERT
- ADD
- GET (ALL)
- GET
- DELETE
- MODIFY
- GEN_CSR
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

inventory

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey?action=inventory;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_certkey": {

"ns_ip_address":<String_value>,

"password":<String_value>,

"ssl_key":<String_value>,

"certkeypair_name":<String_value>,

"save_config":<Boolean_value>,

"file_location_path":<String_value>,

"no_domain_check":<Boolean_value>,

"file_name":<String_value>,

"certkeypair_arr":<String_value[]>,

"id":<String_value>,

"ns_ip_address_arr":<String_value[]>,

"certchainbinding":<String_value[]>,

"source_ipaddress":<String_value>,

"managed_by":<String_value>,

"cert_format":<String_value>,

"certificate_data":<String_value>,

"ssl_certificate":<String_value>,

"snicert":<Boolean_value>,
```

```
"app_name":<String_value>,

"source_certificate":<String_value>,

"certificate_dn":<String_value>,

"key_data":<String_value>,

"entity_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certkey":[{

"serial_number":<String_value>,

"signature_algorithm":<String_value>,

"valid_from":<String_value>,

"status":<String_value>,

"hostname":<String_value>,

"issuer":<String_value>,

"no_domain_check":<Boolean_value>,

"ns_ip_address":<String_value>,

"certkeypair_name":<String_value>,

"certificate_dn":<String_value>,

"public_key_size":<Integer_value>,
```

```
"device_name":<String_value>,

"ssl_key":<String_value>,

"id":<String_value>,

"key_data":<String_value>,

"valid_to":<String_value>,

"subject":<String_value>,

"version":<Integer_value>,

"managed_by":<String_value>,

"public_key_algorithm":<String_value>,

"days_to_expiry":<Integer_value>,

"poll_time":<Integer_value>,

"key_name":<String_value>,

"display_name":<String_value>,

"no_of_bound_entities":<Integer_value>,

"ssl_certificate":<String_value>,

"certificate_data":<String_value>,

"file_location_path":<String_value>,

"partition_name":<String_value>,

"subjaltname":<String_value>,

"cert_format":<String_value>,

"certchainbinding":<String_value>,

"app_name":<String_value>,
```

```
"password":<String_value>,

"source_certificate":<String_value>,

"domain":<String_value>,

"file_name":<String_value>,

"certkeypair_arr":<String_value>,

"source_ipaddress":<String_value>,

"save_config":<Boolean_value>,

"datacenter_id":<String_value>,

"activity_id":<String_value>,

"ns_ip_address_arr":<String_value>,

"entity_name":<String_value>,

"snicert":<Boolean_value>,

"csr":<String_value>}]}
```

list_entity_cert

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey?action=list_entity_cert;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_certkey": {

"ns_ip_address":<String_value>,

"password":<String_value>,

"ssl_key":<String_value>,

"certkeypair_name":<String_value>,

"save_config":<Boolean_value>,

"file_location_path":<String_value>,

"no_domain_check":<Boolean_value>,

"file_name":<String_value>,

"certkeypair_arr":<String_value[]>,

"id":<String_value>,

"ns_ip_address_arr":<String_value[]>,

"certchainbinding":<String_value[]>,

"source_ipaddress":<String_value>,

"managed_by":<String_value>,

"cert_format":<String_value>,

"certificate_data":<String_value>,

"ssl_certificate":<String_value>,

"snicert":<Boolean_value>,

"app_name":<String_value>,

"source_certificate":<String_value>,

"certificate_dn":<String_value>,
```

```
"key_data":<String_value>,

"entity_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certkey":[{

"serial_number":<String_value>,

"signature_algorithm":<String_value>,

"valid_from":<String_value>,

"status":<String_value>,

"hostname":<String_value>,

"issuer":<String_value>,

"no_domain_check":<Boolean_value>,

"ns_ip_address":<String_value>,

"certkeypair_name":<String_value>,

"certificate_dn":<String_value>,

"public_key_size":<Integer_value>,

"device_name":<String_value>,

"ssl_key":<String_value>,

"id":<String_value>,
```

```
"key_data":<String_value>,

"valid_to":<String_value>,

"subject":<String_value>,

"version":<Integer_value>,

"managed_by":<String_value>,

"public_key_algorithm":<String_value>,

"days_to_expiry":<Integer_value>,

"poll_time":<Integer_value>,

"key_name":<String_value>,

"display_name":<String_value>,

"no_of_bound_entities":<Integer_value>,

"ssl_certificate":<String_value>,

"certificate_data":<String_value>,

"file_location_path":<String_value>,

"partition_name":<String_value>,

"subjaltname":<String_value>,

"cert_format":<String_value>,

"certchainbinding":<String_value>,

"app_name":<String_value>,

"password":<String_value>,

"source_certificate":<String_value>,

"domain":<String_value>,
```

```
"file_name":<String_value>,

"certkeypair_arr":<String_value>,

"source_ipaddress":<String_value>,

"save_config":<Boolean_value>,

"datacenter_id":<String_value>,

"activity_id":<String_value>,

"ns_ip_address_arr":<String_value>,

"entity_name":<String_value>,

"snicert":<Boolean_value>,

"csr":<String_value>}]}
```

bind_cert

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey?action=bind_cert;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_certkey": {

"ns_ip_address":<String_value>,

"password":<String_value>,
```

```
"ssl_key":<String_value>,

"certkeypair_name":<String_value>,

"save_config":<Boolean_value>,

"file_location_path":<String_value>,

"no_domain_check":<Boolean_value>,

"file_name":<String_value>,

"certkeypair_arr":<String_value[]>,

"id":<String_value>,

"ns_ip_address_arr":<String_value[]>,

"certchainbinding":<String_value[]>,

"source_ipaddress":<String_value>,

"managed_by":<String_value>,

"cert_format":<String_value>,

"certificate_data":<String_value>,

"ssl_certificate":<String_value>,

"snicert":<Boolean_value>,

"app_name":<String_value>,

"source_certificate":<String_value>,

"certificate_dn":<String_value>,

"key_data":<String_value>,

"entity_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certkey":[{

"serial_number":<String_value>,

"signature_algorithm":<String_value>,

"valid_from":<String_value>,

"status":<String_value>,

"hostname":<String_value>,

"issuer":<String_value>,

"no_domain_check":<Boolean_value>,

"ns_ip_address":<String_value>,

"certkeypair_name":<String_value>,

"certificate_dn":<String_value>,

"public_key_size":<Integer_value>,

"device_name":<String_value>,

"ssl_key":<String_value>,

"id":<String_value>,

"key_data":<String_value>,

"valid_to":<String_value>,
```

```
"subject":<String_value>,

"version":<Integer_value>,

"managed_by":<String_value>,

"public_key_algorithm":<String_value>,

"days_to_expiry":<Integer_value>,

"poll_time":<Integer_value>,

"key_name":<String_value>,

"display_name":<String_value>,

"no_of_bound_entities":<Integer_value>,

"ssl_certificate":<String_value>,

"certificate_data":<String_value>,

"file_location_path":<String_value>,

"partition_name":<String_value>,

"subjaltname":<String_value>,

"cert_format":<String_value>,

"certchainbinding":<String_value>,

"app_name":<String_value>,

"password":<String_value>,

"source_certificate":<String_value>,

"domain":<String_value>,

"file_name":<String_value>,

"certkeypair_arr":<String_value>,
```

```
"source_ipaddress":<String_value>,

"save_config":<Boolean_value>,

"datacenter_id":<String_value>,

"activity_id":<String_value>,

"ns_ip_address_arr":<String_value>,

"entity_name":<String_value>,

"snicert":<Boolean_value>,

"csr":<String_value>}]}
```

add

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_certkey": {

<b>"certkeypair_name":<String_value></b>,

<b>"ns_ip_address_arr":<String_value[]></b>,

"ns_ip_address":<String_value>,

"password":<String_value>,
```

```
"ssl_key":<String_value>,

"save_config":<Boolean_value>,

"file_location_path":<String_value>,

"no_domain_check":<Boolean_value>,

"file_name":<String_value>,

"certkeypair_arr":<String_value[]>,

"id":<String_value>,

"certchainbinding":<String_value[]>,

"source_ipaddress":<String_value>,

"managed_by":<String_value>,

"cert_format":<String_value>,

"certificate_data":<String_value>,

"ssl_certificate":<String_value>,

"snicert":<Boolean_value>,

"app_name":<String_value>,

"source_certificate":<String_value>,

"certificate_dn":<String_value>,

"key_data":<String_value>,

"entity_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certkey":[{

"serial_number":<String_value>,

"signature_algorithm":<String_value>,

"valid_from":<String_value>,

"status":<String_value>,

"hostname":<String_value>,

"issuer":<String_value>,

"no_domain_check":<Boolean_value>,

"ns_ip_address":<String_value>,

"certkeypair_name":<String_value>,

"certificate_dn":<String_value>,

"public_key_size":<Integer_value>,

"device_name":<String_value>,

"ssl_key":<String_value>,

"id":<String_value>,

"key_data":<String_value>,

"valid_to":<String_value>,

"subject":<String_value>,

"version":<Integer_value>,

"managed_by":<String_value>,
```

```
"public_key_algorithm":<String_value>,

"days_to_expiry":<Integer_value>,

"poll_time":<Integer_value>,

"key_name":<String_value>,

"display_name":<String_value>,

"no_of_bound_entities":<Integer_value>,

"ssl_certificate":<String_value>,

"certificate_data":<String_value>,

"file_location_path":<String_value>,

"partition_name":<String_value>,

"subjaltname":<String_value>,

"cert_format":<String_value>,

"certchainbinding":<String_value>,

"app_name":<String_value>,

"password":<String_value>,

"source_certificate":<String_value>,

"domain":<String_value>,

"file_name":<String_value>,

"certkeypair_arr":<String_value>,

"source_ipaddress":<String_value>,

"save_config":<Boolean_value>,

"datacenter_id":<String_value>,
```

```
"activity_id":<String_value>,

"ns_ip_address_arr":<String_value>,

"entity_name":<String_value>,

"snicert":<Boolean_value>,

"csr":<String_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_ssl_certkey resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_ssl_certkey resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey **?count=yes** Use this query-parameter to get the count of ns_ssl_certkey resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certkey":[{

"serial_number":<String_value>,

"signature_algorithm":<String_value>,

"valid_from":<String_value>,

"status":<String_value>,

"hostname":<String_value>,

"issuer":<String_value>,

"no_domain_check":<Boolean_value>,

"ns_ip_address":<String_value>,

"certkeypair_name":<String_value>,

"certificate_dn":<String_value>,

"public_key_size":<Integer_value>,

"device_name":<String_value>,

"ssl_key":<String_value>,

"id":<String_value>,

"key_data":<String_value>,

"valid_to":<String_value>,

"subject":<String_value>,
```

```
"version":<Integer_value>,

"managed_by":<String_value>,

"public_key_algorithm":<String_value>,

"days_to_expiry":<Integer_value>,

"poll_time":<Integer_value>,

"key_name":<String_value>,

"display_name":<String_value>,

"no_of_bound_entities":<Integer_value>,

"ssl_certificate":<String_value>,

"certificate_data":<String_value>,

"file_location_path":<String_value>,

"partition_name":<String_value>,

"subjaltname":<String_value>,

"cert_format":<String_value>,

"certchainbinding":<String_value>,

"app_name":<String_value>,

"password":<String_value>,

"source_certificate":<String_value>,

"domain":<String_value>,

"file_name":<String_value>,

"certkeypair_arr":<String_value>,

"source_ipaddress":<String_value>,
```

```
"save_config":<Boolean_value>,

"datacenter_id":<String_value>,

"activity_id":<String_value>,

"ns_ip_address_arr":<String_value>,

"entity_name":<String_value>,

"snicert":<Boolean_value>,

"csr":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certkey":[{

"serial_number":<String_value>,
```

```
"signature_algorithm":<String_value>,

"valid_from":<String_value>,

"status":<String_value>,

"hostname":<String_value>,

"issuer":<String_value>,

"no_domain_check":<Boolean_value>,

"ns_ip_address":<String_value>,

"certkeypair_name":<String_value>,

"certificate_dn":<String_value>,

"public_key_size":<Integer_value>,

"device_name":<String_value>,

"ssl_key":<String_value>,

"id":<String_value>,

"key_data":<String_value>,

"valid_to":<String_value>,

"subject":<String_value>,

"version":<Integer_value>,

"managed_by":<String_value>,

"public_key_algorithm":<String_value>,

"days_to_expiry":<Integer_value>,

"poll_time":<Integer_value>,

"key_name":<String_value>,
```

```
"display_name":<String_value>,

"no_of_bound_entities":<Integer_value>,

"ssl_certificate":<String_value>,

"certificate_data":<String_value>,

"file_location_path":<String_value>,

"partition_name":<String_value>,

"subjaltname":<String_value>,

"cert_format":<String_value>,

"certchainbinding":<String_value>,

"app_name":<String_value>,

"password":<String_value>,

"source_certificate":<String_value>,

"domain":<String_value>,

"file_name":<String_value>,

"certkeypair_arr":<String_value>,

"source_ipaddress":<String_value>,

"save_config":<Boolean_value>,

"datacenter_id":<String_value>,

"activity_id":<String_value>,

"ns_ip_address_arr":<String_value>,

"entity_name":<String_value>,

"snicert":<Boolean_value>,
```

```
"csr":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

modify

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_certkey":{

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"password":<String_value>,

"ssl_key":<String_value>,

"certkeypair_name":<String_value>,

"save_config":<Boolean_value>,

"file_location_path":<String_value>,

"no_domain_check":<Boolean_value>,

"file_name":<String_value>,

"certkeypair_arr":<String_value[]>,

"ns_ip_address_arr":<String_value[]>,

"certchainbinding":<String_value[]>,

"source_ipaddress":<String_value>,

"managed_by":<String_value>,

"cert_format":<String_value>,

"certificate_data":<String_value>,

"ssl_certificate":<String_value>,

"snicert":<Boolean_value>,

"app_name":<String_value>,

"source_certificate":<String_value>,

"certificate_dn":<String_value>,
```

```
"key_data":<String_value>,

"entity_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certkey":[{

"serial_number":<String_value>,

"signature_algorithm":<String_value>,

"valid_from":<String_value>,

"status":<String_value>,

"hostname":<String_value>,

"issuer":<String_value>,

"no_domain_check":<Boolean_value>,

"ns_ip_address":<String_value>,

"certkeypair_name":<String_value>,

"certificate_dn":<String_value>,

"public_key_size":<Integer_value>,

"device_name":<String_value>,

"ssl_key":<String_value>,

"id":<String_value>,
```

```
"key_data":<String_value>,

"valid_to":<String_value>,

"subject":<String_value>,

"version":<Integer_value>,

"managed_by":<String_value>,

"public_key_algorithm":<String_value>,

"days_to_expiry":<Integer_value>,

"poll_time":<Integer_value>,

"key_name":<String_value>,

"display_name":<String_value>,

"no_of_bound_entities":<Integer_value>,

"ssl_certificate":<String_value>,

"certificate_data":<String_value>,

"file_location_path":<String_value>,

"partition_name":<String_value>,

"subjaltname":<String_value>,

"cert_format":<String_value>,

"certchainbinding":<String_value>,

"app_name":<String_value>,

"password":<String_value>,

"source_certificate":<String_value>,

"domain":<String_value>,
```

```
"file_name":<String_value>,

"certkeypair_arr":<String_value>,

"source_ipaddress":<String_value>,

"save_config":<Boolean_value>,

"datacenter_id":<String_value>,

"activity_id":<String_value>,

"ns_ip_address_arr":<String_value>,

"entity_name":<String_value>,

"snicert":<Boolean_value>,

"csr":<String_value>}]}
```

gen_csr

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey/id_value<String>?action=gen_csr;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_certkey": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

download

URL: https://<MGMT-IP>/nitro/v2/download/;ltns_ip_address;gt_;ltcertkeypair_name;gt.tgz **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload: Binary Stream

ns_ssl_certkey_policy

Configuration for Citrix ADC SSL Cert-Key Polling Policy resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
interval_unit	<String>	Read-write	This property is deprecated, polling interval will be provided in minutes

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			only;ltbr;gtFrequency unit (Minutes).
polling_interval	<Integer>	Read-write	Frequency of polling in minutes. Minimum value = 30 Maximum value =
ns_ip_address	<String[]>	Read-write	List of Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
act_id	<String>	Read-write	Activity ID that is to track the polling status using resource poll_activity_status..

Operations

(click to see Properties)

- DO_POLL
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

do_poll

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey_policy?action=do_poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_certkey_policy": {

<b>"polling_interval":<Integer_value></b>,
```

```
"ns_ip_address":<String_value[]>,

"act_id":<String_value>,

"interval_unit":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certkey_policy":[{

"interval_unit":<String_value>,

"polling_interval":<Integer_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey_policy **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certkey_policy":[{

"interval_unit":<String_value>,

"polling_interval":<Integer_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certkey_policy/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_certkey_policy":{

<b>"polling_interval":<Integer_value></b>,

"ns_ip_address":<String_value[]>,

"act_id":<String_value>,

"interval_unit":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certkey_policy":[{

"interval_unit":<String_value>,

"polling_interval":<Integer_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

ns_ssl_certlink

Configuration for Citrix ADC SSL Certificate Link resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
linkcertkeyname	<String>	Read-write	Certificate Name To Link.
certkey	<String>	Read-write	Certificate Name.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address.

Operations

(click to see Properties)

- UNLINK

- LINK

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

unlink

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certlink?action=unlink;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_certlink": {

"ns_ip_address":<String_value>,

"linkcertkeyname":<String_value>,

"certkey":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certlink":[{

"linkcertkeyname":<String_value>,

"certkey":<String_value>,

"ns_ip_address":<String_value>}]}
```

link

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_certlink?action=link;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_certlink": {

"ns_ip_address":<String_value>,

"linkcertkeyname":<String_value>,

"certkey":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_certlink":[{

"linkcertkeyname":<String_value>,

"certkey":<String_value>,

"ns_ip_address":<String_value>}]}
```

ns_ssl_csr

Configuration for CSR File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
statename	<String>	Read-write	State Name. Minimum length = 2 Maximum length = 128
issuer	<String>	Read-write	Third party Certificate Authority provider.
keyform	<String>	Read-write	Key Form.
keyalgorithm	<String>	Read-write	Key Algorithm.Possible values: RSA, ECC.
app_name	<String>	Read-write	Application name.Used for tracking task logs created for a specific application.
certificate_dn	<String>	Read-write	Applicable only if the issuer is Venafi. Certificate path is populated in this field..
keystrength	<String>	Read-write	Key bits in size, if algorithm is RSA.EllipticCurve, if algorithm is ECC.
city	<String>	Read-write	City.
file_name	<String>	Read-write	Name for and, optionally, path to the certificate signing request (CSR). Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 256
countryname	<String>	Read-write	Country Name. Minimum length = 2 Maximum length = 128
policy	<String>	Read-write	Applicable only if the issuer is Venafi.Policy folder path which determines the certificate attributes..
passphrase	<String>	Read-write	Pass Phrase used to encrypt the private-key.
is_deploy	<Boolean>	Read-write	Whether Auto deployment of certificates managed by third party CA should be done by ADM..
commonname	<String>	Read-write	Common Name. Minimum length = 1
organizationname	<String>	Read-write	Organization Name. Minimum length = 1
organizationunit	<String>	Read-write	Organization Unit.
file_location_path	<String>	Read-write	File Location on Client for download. Minimum length = 1
subjaltname	<String>	Read-write	Subject alternative name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
keyfile	<String>	Read-write	Key File. Minimum length = 1 Maximum length = 128
certificate_name	<String>	Read-write	Name of the certificate.
serial_number	<String>	Read-only	Serial Number.
file_size	<Integer>	Read-only	file_size.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).
tenant_name	<String>	Read-only	Tenant Name.
activity_id	<String>	Read-only	Activity Id used to track the progress of create certificate.
file_last_modified	<String>	Read-only	Last Modified.
csr	<String>	Read-only	Generated CSR.

Operations

(click to see Properties)

- CREATE_CERTIFICATE
- ADD
- RENEW_CERTIFICATE
- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

create_certificate

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_csr?action=create_certificate;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_csr": {

"keystrength":<String_value>,

"commonname":<String_value>,

"organizationname":<String_value>,

"organizationunit":<String_value>,

"file_location_path":<String_value>,

"file_name":<String_value>,

"policy":<String_value>,

"certificate_name":<String_value>,

"city":<String_value>,

"subjaltname":<String_value>,

"keyfile":<String_value>,

"issuer":<String_value>,

"is_deploy":<Boolean_value>,

"keyform":<String_value>,

"keyalgorithm":<String_value>,

"passphrase":<String_value>,

"statename":<String_value>,
```

```
"app_name":<String_value>,

"certificate_dn":<String_value>,

"countryname":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_csr":[{

"serial_number":<String_value>,

"statename":<String_value>,

"file_size":<Integer_value>,

"issuer":<String_value>,

"keyform":<String_value>,

"is_dir":<Boolean_value>,

"keyalgorithm":<String_value>,

"app_name":<String_value>,

"certificate_dn":<String_value>,

"keystrength":<String_value>,

"city":<String_value>,

"file_name":<String_value>,

"countryname":<String_value>,
```

```
"policy":<String_value>,

"passphrase":<String_value>,

"file_last_modified_epoch":<Double_value>,

"is_deploy":<Boolean_value>,

"commonname":<String_value>,

"tenant_name":<String_value>,

"activity_id":<String_value>,

"organizationname":<String_value>,

"organizationunit":<String_value>,

"file_last_modified":<String_value>,

"certificate_policy_object":<certificate_policy_value>,

"file_location_path":<String_value>,

"subjaltname":<String_value>,

"keyfile":<String_value>,

"csr":<String_value>,

"certificate_name":<String_value>}]}
```

add

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_csr?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_csr": {

<b>"statename":<String_value></b>,

<b>"file_name":<String_value></b>,

<b>"countryname":<String_value></b>,

<b>"commonname":<String_value></b>,

<b>"organizationname":<String_value></b>,

<b>"keyfile":<String_value></b>,

"keystrength":<String_value>,

"organizationunit":<String_value>,

"file_location_path":<String_value>,

"policy":<String_value>,

"certificate_name":<String_value>,

"city":<String_value>,

"subjaltname":<String_value>,

"issuer":<String_value>,

"is_deploy":<Boolean_value>,

"keyform":<String_value>,

"keyalgorithm":<String_value>,

"passphrase":<String_value>,
```

```
"app_name":<String_value>,

"certificate_dn":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_csr":[{

"serial_number":<String_value>,

"statename":<String_value>,

"file_size":<Integer_value>,

"issuer":<String_value>,

"keyform":<String_value>,

"is_dir":<Boolean_value>,

"keyalgorithm":<String_value>,

"app_name":<String_value>,

"certificate_dn":<String_value>,

"keystrength":<String_value>,

"city":<String_value>,

"file_name":<String_value>,

"countryname":<String_value>,

"policy":<String_value>,
```

```
"passphrase":<String_value>,

"file_last_modified_epoch":<Double_value>,

"is_deploy":<Boolean_value>,

"commonname":<String_value>,

"tenant_name":<String_value>,

"activity_id":<String_value>,

"organizationname":<String_value>,

"organizationunit":<String_value>,

"file_last_modified":<String_value>,

"certificate_policy_object":<certificate_policy_value>,

"file_location_path":<String_value>,

"subjaltname":<String_value>,

"keyfile":<String_value>,

"csr":<String_value>,

"certificate_name":<String_value>}]}
```

renew_certificate

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_csr?action=renew_certificate;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_csr": {

"keystrength":<String_value>,

"commonname":<String_value>,

"organizationname":<String_value>,

"organizationunit":<String_value>,

"file_location_path":<String_value>,

"file_name":<String_value>,

"policy":<String_value>,

"certificate_name":<String_value>,

"city":<String_value>,

"subjaltname":<String_value>,

"keyfile":<String_value>,

"issuer":<String_value>,

"is_deploy":<Boolean_value>,

"keyform":<String_value>,

"keyalgorithm":<String_value>,

"passphrase":<String_value>,

"statename":<String_value>,

"app_name":<String_value>,

"certificate_dn":<String_value>,
```

```
"countryname":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_csr":[{

"serial_number":<String_value>,

"statename":<String_value>,

"file_size":<Integer_value>,

"issuer":<String_value>,

"keyform":<String_value>,

"is_dir":<Boolean_value>,

"keyalgorithm":<String_value>,

"app_name":<String_value>,

"certificate_dn":<String_value>,

"keystrength":<String_value>,

"city":<String_value>,

"file_name":<String_value>,

"countryname":<String_value>,

"policy":<String_value>,

"passphrase":<String_value>,
```

```
"file_last_modified_epoch":<Double_value>,

"is_deploy":<Boolean_value>,

"commonname":<String_value>,

"tenant_name":<String_value>,

"activity_id":<String_value>,

"organizationname":<String_value>,

"organizationunit":<String_value>,

"file_last_modified":<String_value>,

"certificate_policy_object":<certificate_policy_value>,

"file_location_path":<String_value>,

"subjaltname":<String_value>,

"keyfile":<String_value>,

"csr":<String_value>,

"certificate_name":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_csr/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_csr **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_ssl_csr **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_ssl_csr resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_ssl_csr **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_ssl_csr resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_ssl_csr **?count=yes** Use this query-parameter to get the count of ns_ssl_csr resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_csr":[{

"serial_number":<String_value>,

"statename":<String_value>,

"file_size":<Integer_value>,

"issuer":<String_value>,

"keyform":<String_value>,

"is_dir":<Boolean_value>,

"keyalgorithm":<String_value>,

"app_name":<String_value>,

"certificate_dn":<String_value>,

"keystrength":<String_value>,

"city":<String_value>,

"file_name":<String_value>,

"countryname":<String_value>,

"policy":<String_value>,

"passphrase":<String_value>,

"file_last_modified_epoch":<Double_value>,

"is_deploy":<Boolean_value>,

"commonname":<String_value>,

"tenant_name":<String_value>,
```

```
"activity_id":<String_value>,

"organizationname":<String_value>,

"organizationunit":<String_value>,

"file_last_modified":<String_value>,

"certificate_policy_object":<certificate_policy_value>,

"file_location_path":<String_value>,

"subjaltname":<String_value>,

"keyfile":<String_value>,

"csr":<String_value>,

"certificate_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_csr/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_csr":[{

"serial_number":<String_value>,

"statename":<String_value>,

"file_size":<Integer_value>,

"issuer":<String_value>,

"keyform":<String_value>,

"is_dir":<Boolean_value>,

"keyalgorithm":<String_value>,

"app_name":<String_value>,

"certificate_dn":<String_value>,

"keystrength":<String_value>,

"city":<String_value>,

"file_name":<String_value>,

"countryname":<String_value>,

"policy":<String_value>,

"passphrase":<String_value>,

"file_last_modified_epoch":<Double_value>,

"is_deploy":<Boolean_value>,

"commonname":<String_value>,

"tenant_name":<String_value>,
```

```
"activity_id":<String_value>,

"organizationname":<String_value>,

"organizationunit":<String_value>,

"file_last_modified":<String_value>,

"certificate_policy_object":<certificate_policy_value>,

"file_location_path":<String_value>,

"subjaltname":<String_value>,

"keyfile":<String_value>,

"csr":<String_value>,

"certificate_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/ns_ssl_csr **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

ns_ssl_files

Configuration for Citrix ADC SSL Files resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location of ssl file on ADC. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
id	<String>	Read-write	Id is system generated key for all the ns ssl files.
display_name	<String>	Read-write	display name of device on which ssl file is present.
ns_ip_address	<String>	Read-write	device ip on on which ssl file is present.
act_id	<String>	Read-write	Activity ID that is used by GUI to track the polling status..
file_size	<String>	Read-only	file_size.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_last_modified	<String>	Read-only	Last Modified.

Operations

(click to see Properties)

- DO_POLL
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

do_poll

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_files?action=do_poll;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_files": {

"ns_ip_address":<String_value>,

"act_id":<String_value>,

"file_location_path":<String_value>,

"file_name":<String_value>,

"id":<String_value>,

"display_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_files":[{

"file_location_path":<String_value>,

"file_size":<String_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"id":<String_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_files/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_files **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_ssl_files **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_ssl_files resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_ssl_files **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_ssl_files resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_ssl_files **?count=yes** Use this query-parameter to get the count of ns_ssl_files resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_files":[{

"file_location_path":<String_value>,

"file_size":<String_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,
```

```
"id":<String_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_files/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_files":[{

"file_location_path":<String_value>,

"file_size":<String_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,
```

```
"id":<String_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

download

URL: https://<MGMT-IP>/nitro/v2/download/ns_ssl_files/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload: Binary Stream

ns_ssl_key

Configuration for Citrix ADC SSL key File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
keysize	<Integer>	Read-write	Key Size in bits.
keyform	<String>	Read-write	Key Form.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
des_type	<String>	Read-write	DES Encryption type possible values des| des3..
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
password	<String>	Read-write	Password.
algo	<String>	Read-write	Encryption Algorithm supported RSA and DSA..
file_name	<String>	Read-write	Key File Name. Minimum length = 1 Maximum length = 256
exponent	<String>	Read-write	Exponent.
ec_curve	<String>	Read-write	Elliptic curve.
tenant_name	<String>	Read-only	Tenant Name.
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- CREATE
- UPLOAD
- DELETE

- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

create

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_key?action=create;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ssl_key": {

"password":<String_value>,

"file_location_path":<String_value>,

"file_name":<String_value>,

"des_type":<String_value>,

"ec_curve":<String_value>,

"keysize":<Integer_value>,

"keyform":<String_value>,

"algo":<String_value>,

"exponent":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_key":[{

"tenant_name":<String_value>,
```

```
"keysize":<Integer_value>,

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"keyform":<String_value>,

"is_dir":<Boolean_value>,

"des_type":<String_value>,

"file_location_path":<String_value>,

"password":<String_value>,

"algo":<String_value>,

"file_name":<String_value>,

"exponent":<String_value>,

"ec_curve":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_key/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_key **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_ssl_key **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_ssl_key resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_ssl_key **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_ssl_key resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_ssl_key **?count=yes** Use this query-parameter to get the count of ns_ssl_key resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_key":[{

"tenant_name":<String_value>,

"keysize":<Integer_value>,

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"keyform":<String_value>,

"is_dir":<Boolean_value>,

"des_type":<String_value>,

"file_location_path":<String_value>,

"password":<String_value>,

"algo":<String_value>,

"file_name":<String_value>,

"exponent":<String_value>,

"ec_curve":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_ssl_key/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ssl_key":[{

"tenant_name":<String_value>,

"keysize":<Integer_value>,

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"keyform":<String_value>,

"is_dir":<Boolean_value>,

"des_type":<String_value>,

"file_location_path":<String_value>,

"password":<String_value>,

"algo":<String_value>,

"file_name":<String_value>,

"exponent":<String_value>,

"ec_curve":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

```
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/ns_ssl_key **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

ssl_cert_threshold

Configuration for SSL certificate Threshold configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	name.
region	<String>	Read-write	Region Code.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 256
rule	<String>	Read-write	Rule.
stylebook_flag	<Boolean>	Read-write	Stylebook threshold configuration true or false.
country	<String>	Read-write	Country Code. Maximum length = 256
sms_profile	<String>	Read-write	SMS Profile.
slack_profile	<String>	Read-write	Slack Profile.
feature	<String>	Read-write	feature.
ctnsappname	<String>	Read-write	AppName.
hide_default_certs	<Boolean>	Read-write	Set the value to true if default certs in device need not be managed.
id	<String>	Read-write	Id is system generated key for all the Thresholds configuration . Maximum length = 256
resource_type	<String>	Read-write	Resource Type.
mail_profile	<String>	Read-write	Mail Profile.
is_enabled	<String>	Read-write	true or false.
pagerduty_profile	<String>	Read-write	PagerDuty Profile.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
duration	<String>	Read-write	duration of metric to be checked against threshold.
days_to_expiry	<Integer>	Read-write	Days remaining for expiry. Maximum value =
servicenow_profile	<String>	Read-write	Servicenow Profile.
city	<String>	Read-write	City. Maximum length = 256
ip_address	<String>	Read-write	Field to store the ip address as it is along with any extension. Maximum length = 512
group_name	<String>	Read-write	Group Name.
reference_key	<String>	Read-write	Reference Key.
exporter_id	<Double>	Read-write	Exporter Id.
approved_key_strength_list	<String[]>	Read-write	Key Strength List.
approved_issuers_list	<String[]>	Read-write	Issuers List.
approved_ssl_protocol_list	<String[]>	Read-write	SSL Protocols List.
approved_signature_algo_list	<String[]>	Read-write	Signature Algorithm List.
tenant_name	<String>	Read-only	Tenant Name.

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ssl_cert_threshold **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ssl_cert_threshold **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ssl_cert_threshold resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ssl_cert_threshold **?pagesize=#no;pageno=#no** Use this query-parameter to get the ssl_cert_threshold resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ssl_cert_threshold **?count=yes** Use this query-parameter to get the count of ssl_cert_threshold resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ssl_cert_threshold":[{

"name":<String_value>,

"region":<String_value>,
```

```
"rule":<String_value>,

"stylebook_flag":<Boolean_value>,

"country":<String_value>,

"issuers_list":<String_value>,

"sms_profile":<String_value>,

"signature_algo_list":<String_value>,

"slack_profile":<String_value>,

"feature":<String_value>,

"ctnsappname":<String_value>,

"hide_default_certs":<Boolean_value>,

"id":<String_value>,

"resource_type":<String_value>,

"mail_profile":<String_value>,

"is_enabled":<String_value>,

"pagerduty_profile":<String_value>,

"duration":<String_value>,

"days_to_expiry":<Integer_value>,

"servicenow_profile":<String_value>,

"city":<String_value>,

"key_strength_list":<String_value>,

"ip_address":<String_value>,

"ssl_protocol_list":<String_value>,
```

```
"tenant_name":<String_value>,

"group_name":<String_value>,

"reference_key":<String_value>,

"exporter_id":<Double_value>,

"approved_key_strength_list":<String_value>,

"approved_issuers_list":<String_value>,

"approved_ssl_protocol_list":<String_value>,

"approved_signature_algo_list":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ssl_cert_threshold/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ssl_cert_threshold":[{

"name":<String_value>,

"region":<String_value>,

"rule":<String_value>,

"stylebook_flag":<Boolean_value>,

"country":<String_value>,

"issuers_list":<String_value>,

"sms_profile":<String_value>,

"signature_algo_list":<String_value>,

"slack_profile":<String_value>,

"feature":<String_value>,

"ctnsappname":<String_value>,

"hide_default_certs":<Boolean_value>,

"id":<String_value>,

"resource_type":<String_value>,

"mail_profile":<String_value>,

"is_enabled":<String_value>,

"pagerduty_profile":<String_value>,

"duration":<String_value>,

"days_to_expiry":<Integer_value>,

"servicenow_profile":<String_value>,
```

```
"city":<String_value>,

"key_strength_list":<String_value>,

"ip_address":<String_value>,

"ssl_protocol_list":<String_value>,

"tenant_name":<String_value>,

"group_name":<String_value>,

"reference_key":<String_value>,

"exporter_id":<Double_value>,

"approved_key_strength_list":<String_value>,

"approved_issuers_list":<String_value>,

"approved_ssl_protocol_list":<String_value>,

"approved_signature_algo_list":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/ssl_cert_threshold/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ssl_cert_threshold":{

"sms_profile":<String_value>,

"slack_profile":<String_value>,

"feature":<String_value>,

"hide_default_certs":<Boolean_value>,

"ctnsappname":<String_value>,

"id":<String_value>,

"mail_profile":<String_value>,

"resource_type":<String_value>,

"is_enabled":<String_value>,

"pagerduty_profile":<String_value>,

"region":<String_value>,

"name":<String_value>,

"days_to_expiry":<Integer_value>,

"duration":<String_value>,

"approved_issuers_list":<String_value[]>,

"servicenow_profile":<String_value>,

"stylebook_flag":<Boolean_value>,

"rule":<String_value>,

"city":<String_value>,

"ip_address":<String_value>,

"country":<String_value>,
```

```
"group_name":<String_value>,

"approved_ssl_protocol_list":<String_value[]>,

"reference_key":<String_value>,

"approved_key_strength_list":<String_value[]>,

"exporter_id":<Double_value>,

"approved_signature_algo_list":<String_value[]>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ssl_cert_threshold":[{

"name":<String_value>,

"region":<String_value>,

"rule":<String_value>,

"stylebook_flag":<Boolean_value>,

"country":<String_value>,

"issuers_list":<String_value>,

"sms_profile":<String_value>,

"signature_algo_list":<String_value>,

"slack_profile":<String_value>,

"feature":<String_value>,
```

```
"ctnsappname":<String_value>,

"hide_default_certs":<Boolean_value>,

"id":<String_value>,

"resource_type":<String_value>,

"mail_profile":<String_value>,

"is_enabled":<String_value>,

"pagerduty_profile":<String_value>,

"duration":<String_value>,

"days_to_expiry":<Integer_value>,

"servicenow_profile":<String_value>,

"city":<String_value>,

"key_strength_list":<String_value>,

"ip_address":<String_value>,

"ssl_protocol_list":<String_value>,

"tenant_name":<String_value>,

"group_name":<String_value>,

"reference_key":<String_value>,

"exporter_id":<Double_value>,

"approved_key_strength_list":<String_value>,

"approved_issuers_list":<String_value>,

"approved_ssl_protocol_list":<String_value>,

"approved_signature_algo_list":<String_value>}]}
```

Events

- active_event
- command_execution_profile
- device_event_summary
- event
- event_category
- event_cb_device_report
- event_cbsdx_device_report
- event_cbsdx_resrc_threshold_violation_report
- event_config_changes_report
- event_device_report
- event_device_summary
- event_entity_state_changes_report
- event_filter
- event_hardware_failure_report
- event_login_failure_report
- event_ns_device_report
- event_nssdx_device_report
- event_nssdx_resrc_threshold_violation_report
- event_persistant_stat_info
- event_purge_setting
- event_resrc_threshold_violation_report
- event_severity_report
- event_summary
- eventHistory
- fault_category
- fault_failure_object

- fault_severity
- filter_action
- filter_action_props
- filter_criteria
- filter_schedule_details
- filter_scheduler
- ns_event
- ns_event_archive
- ns_event_message
- ns_event_summary
- poll_activity_status
- syslog_filter
- syslog_generic
- syslog_severity
- trap_settings

active_event

Configuration for Active Event resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the events.
severity	<String>	Read-write	Severity of Event.
counter_threshold_value	<String>	Read-only	device threshold value for any threadhold violated traps.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
source	<String>	Read-only	Source from where the event is originated.
cmd_exec_status	<String>	Read-only	Command Execution Status if the event is generated for any config change or config save in device.
device_entity_type	<String>	Read-only	Device Entity Type.
entity	<String>	Read-only	Entity of Event.
device_type	<String>	Read-only	Type of device from which we have received the event.
timestamp	<Integer>	Read-only	Time when the event occured.
operation_type	<String>	Read-only	Operation Type.
category	<String>	Read-only	Category of Event.
failureobj	<String>	Read-only	Failure objects are entity instances or counters for which an event has been generated, for example entity name.
device_family	<String>	Read-only	Family of device from which we have received the event.
message	<String>	Read-only	Event Message - All the other properties recevied in event trap which do not

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			specific fields in this resource(In key : value pair).
user_name	<String>	Read-only	Username.
config_cmd	<String>	Read-only	Config Command if the event is generated for any config change in device.
cmd_auth_status	<String>	Read-only	Command Authorization Status if the event is generated for any config change or config save in device.
device_entity_name	<String>	Read-only	Device Entity Name.
counter_actual_value	<String>	Read-only	device actual value for any threadhold violated traps.
ids_list_arr	<String[]>	Read-only	List of IDS.
current_value	<String>	Read-only	Current Value, can be used while sending traps.
threshold_value	<String>	Read-only	Threshold Value, cab be used while sending traps.
entity_type	<String>	Read-only	Entity Type.
hostname	<String>	Read-only	Assign hostname to managed device, if this is not provided,

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			name will be set as host name .

Operations

(click to see Properties)

- UPDATE_STATUS
- DELETE
- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

update_status

URL: https://<MGMT-IP>/nitro/v2/config/active_event?action=update_status;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"active_event": {

"id":<String_value>,

"severity":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"active_event":[{

"counter_threshold_value":<String_value>,

"source":<String_value>,

"cmd_exec_status":<String_value>,
```

```
"device_entity_type":<String_value>,

"entity":<String_value>,

"device_type":<String_value>,

"timestamp":<Integer_value>,

"operation_type":<String_value>,

"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"failureobj":<String_value>,

"device_family":<String_value>,

"message":<String_value>,

"user_name":<String_value>,

"config_cmd":<String_value>,

"cmd_auth_status":<String_value>,

"device_entity_name":<String_value>,

"counter_actual_value":<String_value>,

"ids_list_arr":<String_value>,

"current_value":<String_value>,

"threshold_value":<String_value>,

"entity_type":<String_value>,

"hostname":<String_value>,

"matched_filters":<String_value>,
```

```
"managed_device_id":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/active_event/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/active_event **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/active_event **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of active_event resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/active_event **? pagesize=#no;pageno=#no** Use this query-parameter to get the active_event resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/active_event **?count=yes** Use this query-parameter to get the count of active_event resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"active_event":[{

"counter_threshold_value":<String_value>,

"source":<String_value>,

"cmd_exec_status":<String_value>,

"device_entity_type":<String_value>,

"entity":<String_value>,

"device_type":<String_value>,

"timestamp":<Integer_value>,

"operation_type":<String_value>,

"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"failureobj":<String_value>,
```

```
"device_family":<String_value>,

"message":<String_value>,

"user_name":<String_value>,

"config_cmd":<String_value>,

"cmd_auth_status":<String_value>,

"device_entity_name":<String_value>,

"counter_actual_value":<String_value>,

"ids_list_arr":<String_value>,

"current_value":<String_value>,

"threshold_value":<String_value>,

"entity_type":<String_value>,

"hostname":<String_value>,

"matched_filters":<String_value>,

"managed_device_id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/active_event/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"active_event":[{

"counter_threshold_value":<String_value>,

"source":<String_value>,

"cmd_exec_status":<String_value>,

"device_entity_type":<String_value>,

"entity":<String_value>,

"device_type":<String_value>,

"timestamp":<Integer_value>,

"operation_type":<String_value>,

"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"failureobj":<String_value>,

"device_family":<String_value>,

"message":<String_value>,

"user_name":<String_value>,
```

```
"config_cmd":<String_value>,

"cmd_auth_status":<String_value>,

"device_entity_name":<String_value>,

"counter_actual_value":<String_value>,

"ids_list_arr":<String_value>,

"current_value":<String_value>,

"threshold_value":<String_value>,

"entity_type":<String_value>,

"hostname":<String_value>,

"matched_filters":<String_value>,

"managed_device_id":<String_value>}]}
```

command_execution_profile

Configuration for Command Execution Profile for Event Rule resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
append_errors	<Boolean>	Read-write	Errors to be appended.
id	<String>	Read-write	Id is system generated key for all the run command profile..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
append_output	<Boolean>	Read-write	Output to be appended.
command_text	<String>	Read-write	Run Command Text. Minimum length = 1 Maximum length = 128
profile_name	<String>	Read-write	Profile name for the run command setting.. Minimum length = 1 Maximum length = 128

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/command_execution_profile?onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"command_execution_profile": {

<b>"command_text":<String_value></b>,
```

```
<b>"profile_name":<String_value></b>,

"id":<String_value>,

"append_errors":<Boolean_value>,

"append_output":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"command_execution_profile":[{

"tenant_id":<String_value>,

"append_errors":<Boolean_value>,

"id":<String_value>,

"append_output":<Boolean_value>,

"command_text":<String_value>,

"profile_name":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/command_execution_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/command_execution_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/command_execution_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of command_execution_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/command_execution_profile **?pagesize=#no;pageno=#no** Use this query-parameter to get the command_execution_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/command_execution_profile **?count=yes** Use this query-parameter to get the count of command_execution_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"command_execution_profile":[{

"tenant_id":<String_value>,

"append_errors":<Boolean_value>,

"id":<String_value>,

"append_output":<Boolean_value>,

"command_text":<String_value>,

"profile_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/command_execution_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"command_execution_profile":[{

"tenant_id":<String_value>,

"append_errors":<Boolean_value>,

"id":<String_value>,

"append_output":<Boolean_value>,

"command_text":<String_value>,

"profile_name":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/command_execution_profile/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"command_execution_profile":{

<b>"id":<String_value></b>,

"append_errors":<Boolean_value>,

"command_text":<String_value>,

"append_output":<Boolean_value>,

"profile_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"command_execution_profile":[{

"tenant_id":<String_value>,

"append_errors":<Boolean_value>,

"id":<String_value>,

"append_output":<Boolean_value>,

"command_text":<String_value>,

"profile_name":<String_value>}]}
```

device_event_summary

Configuration for devices events summary resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key.
device_ip_address	<String>	Read-write	ip_address of the device..
instance_id	<String>	Read-write	Instance ID of the managed device..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 256
type	<String>	Read-write	Device Family. Minimum length = 1 Maximum length = 64
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256
total_critical_events	<Integer>	Read-only	Total number of Critical events city wise..
total_major_events	<Integer>	Read-only	Total number of major events city wise..
total_events	<Integer>	Read-only	Total number of events.
instance_state	<String>	Read-only	State of device, UP only if device accessible.

Operations

(click to see Properties)

- GET (ALL)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/device_event_summary **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/device_event_summary **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the

filtered set of device_event_summary resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/device_event_summary **?pagesize=#no;pageno=#no** Use this query-parameter to get the device_event_summary resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/device_event_summary **?count=yes** Use this query-parameter to get the count of device_event_summary resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_event_summary":[{

"total_critical_events":<Integer_value>,

"total_major_events":<Integer_value>,

"id":<String_value>,

"device_ip_address":<String_value>,

"instance_id":<String_value>,
```

```
"total_events":<Integer_value>,

"type":<String_value>,

"hostname":<String_value>,

"instance_state":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/device_event_summary/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_event_summary":[{

"total_critical_events":<Integer_value>,

"total_major_events":<Integer_value>,

"id":<String_value>,
```

```
"device_ip_address":<String_value>,

"instance_id":<String_value>,

"total_events":<Integer_value>,

"type":<String_value>,

"hostname":<String_value>,

"instance_state":<String_value>}]}
```

event

Configuration for Event resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the events.
severity	<String>	Read-write	Severity of Event. Minimum length = 1 Maximum length = 32
event_history	<eventHistory[]>	Read-write	previous status of event.
counter_threshold_value	<String>	Read-only	device threshold value for any threadhold violated traps.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
source	<String>	Read-only	Source from where the event is generated.
cmd_exec_status	<String>	Read-only	Command Execution Status if the event is generated for any config change or config save in device.
device_entity_type	<String>	Read-only	Device Entity Type.
entity	<String>	Read-only	Entity of Event.
device_type	<String>	Read-only	Type of device for which we have received the event.
timestamp	<Integer>	Read-only	Time when the event is received.
operation_type	<String>	Read-only	Operation Type of the event.
category	<String>	Read-only	Category of Event.
failureobj	<String>	Read-only	Failure objects are entity instances or counters for which an event has been generated.
source_event_id	<Integer>	Read-only	Internal Event ID used in the source device..
history	<String>	Read-only	History of the Event with same entity.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
device_family	<String>	Read-only	Family of device for which we have received the event.
rpt_sample_time	<Double>	Read-only	Report Sample time..
message	<String>	Read-only	Event Message.
user_name	<String>	Read-only	Username.
device_entity_name	<String>	Read-only	Device Entity Name.
config_cmd	<String>	Read-only	Config Command if the event is generated for any config change in device.
cmd_auth_status	<String>	Read-only	Command Authorization Status if the event is generated for any config change or config save in device.
counter_actual_value	<String>	Read-only	device actual value for any threadhold violated traps.
current_value	<String>	Read-only	Current Value, cab be used while sending traps.
threshold_value	<String>	Read-only	Threshold Value, cab be used while sending traps.
entity_type	<String>	Read-only	Entity Type.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hostname	<String>	Read-only	Assign hostname to managed device, if this is not provided, name will be set as host name .
managed_device_id	<String>	Read-only	Managed Device ID.

Operations

(click to see Properties)

- DELETE
- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/event/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event **? pagesize=#no;pageno=#no** Use this query-parameter to get the event resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event **?count=yes** Use this query-parameter to get the count of event resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event":[{

"counter_threshold_value":<String_value>,

"source":<String_value>,

"cmd_exec_status":<String_value>,

"device_entity_type":<String_value>,

"trap_id":<Integer_value>,

"entity":<String_value>,

"device_type":<String_value>,

"timestamp":<Integer_value>,
```

```
"operation_type":<String_value>,

"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"failureobj":<String_value>,

"source_event_id":<Integer_value>,

"event_history":[{

"timestamp":<Integer_value>,

"parent_name":<String_value>,

"id":<String_value>,

"severity":<String_value>,

"message":<String_value>,

"parent_id":<String_value>}],

"history":<String_value>,

"device_family":<String_value>,

"rpt_sample_time":<Double_value>,

"message":<String_value>,

"user_name":<String_value>,

"device_entity_name":<String_value>,

"config_cmd":<String_value>,

"cmd_auth_status":<String_value>,

"counter_actual_value":<String_value>,
```

```
"current_value":<String_value>,

"threshold_value":<String_value>,

"entity_type":<String_value>,

"hostname":<String_value>,

"matched_filters":<String_value>,

"managed_device_id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/event/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event":[{

"counter_threshold_value":<String_value>,

"source":<String_value>,
```

```
"cmd_exec_status":<String_value>,

"device_entity_type":<String_value>,

"trap_id":<Integer_value>,

"entity":<String_value>,

"device_type":<String_value>,

"timestamp":<Integer_value>,

"operation_type":<String_value>,

"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"failureobj":<String_value>,

"source_event_id":<Integer_value>,

"event_history":[{

"timestamp":<Integer_value>,

"parent_name":<String_value>,

"id":<String_value>,

"severity":<String_value>,

"message":<String_value>,

"parent_id":<String_value>}],

"history":<String_value>,

"device_family":<String_value>,

"rpt_sample_time":<Double_value>,
```

```
"message":<String_value>,

"user_name":<String_value>,

"device_entity_name":<String_value>,

"config_cmd":<String_value>,

"cmd_auth_status":<String_value>,

"counter_actual_value":<String_value>,

"current_value":<String_value>,

"threshold_value":<String_value>,

"entity_type":<String_value>,

"hostname":<String_value>,

"matched_filters":<String_value>,

"managed_device_id":<String_value>}]}
```

eventHistory

Configuration for Event History resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_name	<String>	Read-write	.
id	<String>	Read-write	.
parent_id	<String>	Read-write	.
timestamp	<Integer>	Read-only	Event Time.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
severity	<String>	Read-only	Severity of Event.
message	<String>	Read-only	Event Message.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

event_category

Configuration for event category resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
action	<String>	Read-write	Event category action.
category	<String>	Read-write	Event category. Maximum length = 256
display_name	<String>	Read-write	Event category display name. Maximum length = 256
product_type	<String>	Read-only	Product type.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_category **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_category **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_category resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_category **? pagesize=#no;pageno=#no** Use this query-parameter to get the event_category resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_category **?count=yes** Use this query-parameter to get the count of event_category resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_category":[{

"product_type":<String_value>,

"action":<String_value>,

"category":<String_value>,

"display_name":<String_value>}]}
```

event_cb_device_report

Configuration for Event Mgmt Device Instability Reportfor Cloud Bridge Devices resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
severity	<String>	Read-write	severity. Maximum length = 255
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_cb_device_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_cb_device_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_cb_device_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_cb_device_report **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_cb_device_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_cb_device_report **?count=yes** Use this query-parameter to get the count of event_cb_device_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_cb_device_report":[{

"ip_address":<String_value>,

"severity":<String_value>,

"total_count":<Integer_value>,

"hostname":<String_value>}]}
```

event_cbsdx_device_report

Configuration for Event Mgmt Device Instability Report for CBSDX resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
severity	<String>	Read-write	severity. Maximum length = 255
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_cbsdx_device_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_cbsdx_device_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_cbsdx_device_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_cbsdx_device_report **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_cbsdx_device_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_cbsdx_device_report **?count=yes** Use this query-parameter to get the count of event_cbsdx_device_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_cbsdx_device_report":[{

"ip_address":<String_value>,

"severity":<String_value>,

"total_count":<Integer_value>,

"hostname":<String_value>}]}
```

event_cbsdx_resrc_threshold_violation_report

Configuration for Event Mgmt Resrc Threshold Violation Report for CBSDX resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
category	<String>	Read-write	category. Maximum length = 255

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1 Maximum length = 64
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_cbsdx_resrc_threshold_violation_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_cbsdx_resrc_threshold_violation_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_cbsdx_resrc_threshold_violation_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_cbsdx_resrc_threshold_violation_report **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_cbsdx_resrc_threshold_violation_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_cbsdx_resrc_threshold_violation_report **?count=yes** Use this query-parameter to get the count of event_cbsdx_resrc_threshold_violation_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_cbsdx_resrc_threshold_violation_report":[{

"category":<String_value>,

"ip_address":<String_value>,

"total_count":<Integer_value>,

"hostname":<String_value>}]}
```

event_config_changes_report

Configuration for Event Mgmt Config Changes Report resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..
user_name	<String>	Read-write	Username. Maximum length = 255
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_config_changes_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_config_changes_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_config_changes_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_config_changes_report **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_config_changes_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_config_changes_report **?count=yes** Use this query-parameter to get the count of event_config_changes_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_config_changes_report":[{

"ip_address":<String_value>,

"total_count":<Integer_value>,

"user_name":<String_value>,

"hostname":<String_value>}]}
```

event_device_report

Configuration for Event Mgmt Device Instability Report resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
severity	<String>	Read-write	severity. Maximum length = 255
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_device_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_device_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_device_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_device_report **? pagesize=#no;pageno=#no** Use this query-parameter to get the event_device_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_device_report **?count=yes** Use this query-parameter to get the count of event_device_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_device_report":[{

"ip_address":<String_value>,

"severity":<String_value>,

"total_count":<Integer_value>,

"hostname":<String_value>}]}
```

event_device_summary

Configuration for Event Mgmt Device Report resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
category	<String>	Read-write	category.
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
severity	<String>	Read-write	severity. Maximum length = 255
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_device_summary **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_device_summary **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_device_summary resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_device_summary **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_device_summary resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_device_summary **?count=yes** Use this query-parameter to get the count of event_device_summary resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_device_summary":[{

"category":<String_value>,

"ip_address":<String_value>,

"severity":<String_value>,

"total_count":<Integer_value>}]}
```

event_entity_state_changes_report

Configuration for Event Mgmt Entity State Changes Report resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
category	<String>	Read-write	category. Maximum length = 255
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1 Maximum length = 64
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_entity_state_changes_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_entity_state_changes_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_entity_state_changes_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_entity_state_changes_report **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_entity_state_changes_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_entity_state_changes_report **?count=yes** Use this query-parameter to get the count of event_entity_state_changes_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_entity_state_changes_report":[{

"category":<String_value>,

"ip_address":<String_value>,

"total_count":<Integer_value>,

"hostname":<String_value>}]}
```

event_filter

Configuration for event filter criteria properties resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_enabled	<Boolean>	Read-write	enabled or disabled.
actions	<filter_action[]>	Read-write	Filter actions.
event_age_threshold	<Integer>	Read-write	Event Age Threshold - Specify the time interval for which the event has to be in the same state to trigger the action.
name	<String>	Read-write	Filter Name. Maximum length = 255
id	<String>	Read-write	Id is system generated key.
criteria	<filter_criteria>	Read-write	Filter criteria props.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
enable_advanced_regex	<Boolean>	Read-write	Enable Advanced regex for Failure Objects, Message and configuration command.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/event_filter?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"event_filter": {

"event_age_threshold":<Integer_value>,

"id":<String_value>,

"criteria":<filter_criteria_value>,

"is_enabled":<Boolean_value>,

"name":<String_value>,

"actions":[{

"parent_id":<String_value>,

"id":<String_value>,
```

```
"suppress_time":<Integer_value>,

"repeat_email_notification_threshold":<Integer_value>,

"attachment":<String_value>,

"parent_name":<String_value>,

"subject":<String_value>,

"type":<String_value>,

"user_message":<String_value>,

"profile_name":<String_value>}],

"enable_advanced_regex":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_filter":[{

"is_enabled":<Boolean_value>,

"actions":[{

"parent_name":<String_value>,

"props":[{

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,
```

```
"id":<String_value>,

"parent_id":<String_value>}],

"action_name":<String_value>,

"type":<String_value>,

"id":<String_value>,

"suppress_time":<Integer_value>,

"parent_id":<String_value>,

"repeat_email_notification_threshold":<Integer_value>

"attachment":<String_value>

"subject":<String_value>

"sender":<String_value>

"subject_attach_failureobject":<Boolean_value>

"user_message":<String_value>

"profile_name":<String_value>}],

"event_age_threshold":<Integer_value>,

"name":<String_value>,

"id":<String_value>,

"criteria":<filter_criteria_value>,

"enable_advanced_regex":<Boolean_value>,

"is_suppress_enabled":<Boolean_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/event_filter/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_filter **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_filter **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_filter resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_filter **? pagesize=#no;pageno=#no** Use this query-parameter to get the event_filter resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_filter **?count=yes** Use this query-parameter to get the count of event_filter resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_filter":[{

"is_enabled":<Boolean_value>,

"actions":[{

"parent_name":<String_value>,

"props":[{

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"id":<String_value>,

"parent_id":<String_value>}],

"action_name":<String_value>,

"type":<String_value>,

"id":<String_value>,

"suppress_time":<Integer_value>,

"parent_id":<String_value>,
```

```
"repeat_email_notification_threshold":<Integer_value>

"attachment":<String_value>

"subject":<String_value>

"sender":<String_value>

"subject_attach_failureobject":<Boolean_value>

"user_message":<String_value>

"profile_name":<String_value>}],

"event_age_threshold":<Integer_value>,

"name":<String_value>,

"id":<String_value>,

"criteria":<filter_criteria_value>,

"enable_advanced_regex":<Boolean_value>,

"is_suppress_enabled":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/event_filter/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_filter":[{

"is_enabled":<Boolean_value>,

"actions":[{

"parent_name":<String_value>,

"props":[{

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"id":<String_value>,

"parent_id":<String_value>}],

"action_name":<String_value>,

"type":<String_value>,

"id":<String_value>,

"suppress_time":<Integer_value>,

"parent_id":<String_value>,

"repeat_email_notification_threshold":<Integer_value>
```

```
"attachment":<String_value>

"subject":<String_value>

"sender":<String_value>

"subject_attach_failureobject":<Boolean_value>

"user_message":<String_value>

"profile_name":<String_value>}],

"event_age_threshold":<Integer_value>,

"name":<String_value>,

"id":<String_value>,

"criteria":<filter_criteria_value>,

"enable_advanced_regex":<Boolean_value>,

"is_suppress_enabled":<Boolean_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/event_filter/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"event_filter":{

<b>"id":<String_value></b>,

"event_age_threshold":<Integer_value>,

"criteria":<filter_criteria_value>,

"is_enabled":<Boolean_value>,

"name":<String_value>,

"actions":[{

"parent_id":<String_value>,

"id":<String_value>,

"suppress_time":<Integer_value>,

"repeat_email_notification_threshold":<Integer_value>,

"attachment":<String_value>,

"parent_name":<String_value>,

"subject":<String_value>,

"type":<String_value>,

"user_message":<String_value>,

"profile_name":<String_value>}],

"enable_advanced_regex":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_filter":[{

"is_enabled":<Boolean_value>,

"actions":[{

"parent_name":<String_value>,

"props":[{

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"id":<String_value>,

"parent_id":<String_value>}],

"action_name":<String_value>,

"type":<String_value>,

"id":<String_value>,

"suppress_time":<Integer_value>,

"parent_id":<String_value>,

"repeat_email_notification_threshold":<Integer_value>

"attachment":<String_value>

"subject":<String_value>

"sender":<String_value>

"subject_attach_failureobject":<Boolean_value>

"user_message":<String_value>
```

```
"profile_name":<String_value>}],

"event_age_threshold":<Integer_value>,

"name":<String_value>,

"id":<String_value>,

"criteria":<filter_criteria_value>,

"enable_advanced_regex":<Boolean_value>,

"is_suppress_enabled":<Boolean_value>}]}
```

event_hardware_failure_report

Configuration for Event Mgmt Hardware Failure Report resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
category	<String>	Read-write	category. Maximum length = 255
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1 Maximum length = 64
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..
hostname	<String>	Read-write	Assign hostname to managed device, if

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_hardware_failure_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_hardware_failure_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_hardware_failure_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_hardware_failure_report **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_hardware_failure_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_hardware_failure_report **?count=yes** Use this query-parameter to get the count of event_hardware_failure_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_hardware_failure_report":[{

"category":<String_value>,

"ip_address":<String_value>,

"total_count":<Integer_value>,

"hostname":<String_value>}]}
```

event_login_failure_report

Configuration for Event Mgmt Login Failure Report resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1 Maximum length = 64
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..
user_name	<String>	Read-write	Username. Maximum length = 255
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name .

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_login_failure_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_login_failure_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_login_failure_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_login_failure_report **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_login_failure_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_login_failure_report **?count=yes** Use this query-parameter to get the count of event_login_failure_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_login_failure_report":[{
```

```
"ip_address":<String_value>,

"total_count":<Integer_value>,

"user_name":<String_value>,

"hostname":<String_value>}]}
```

event_ns_device_report

Configuration for Event Mgmt Device Instability Report for Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1 Maximum length = 64
severity	<String>	Read-write	severity. Maximum length = 255
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_ns_device_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_ns_device_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_ns_device_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_ns_device_report **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_ns_device_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_ns_device_report **?count=yes** Use this query-parameter to get the count of event_ns_device_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_ns_device_report":[{

"ip_address":<String_value>,
```

```
"severity":<String_value>,

"total_count":<Integer_value>,

"hostname":<String_value>}]}
```

event_nssdx_device_report

Configuration for Event Mgmt Device Instability Report for NSSDX resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1 Maximum length = 64
severity	<String>	Read-write	severity. Maximum length = 255
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_nssdx_device_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_nssdx_device_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_nssdx_device_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_nssdx_device_report **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_nssdx_device_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_nssdx_device_report **?count=yes** Use this query-parameter to get the count of event_nssdx_device_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_nssdx_device_report":[{

"ip_address":<String_value>,

"severity":<String_value>,

"total_count":<Integer_value>,

"hostname":<String_value>}]}
```

event_nssdx_resrc_threshold_violation_report

Configuration for Event Mgmt Resrc Threshold Violation Report for NSSDX resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
category	<String>	Read-write	category. Maximum length = 255
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1 Maximum length = 64
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_nssdx_resrc_threshold_violation_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_nssdx_resrc_threshold_violation_report **? filter=property-name1:property-value1,property-name2:property-value2** Use this query-

parameter to get the filtered set of event_nssdx_resrc_threshold_violation_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_nssdx_resrc_threshold_violation_report **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_nssdx_resrc_threshold_violation_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_nssdx_resrc_threshold_violation_report **?count=yes** Use this query-parameter to get the count of event_nssdx_resrc_threshold_violation_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_nssdx_resrc_threshold_violation_report":[{

"category":<String_value>,

"ip_address":<String_value>,

"total_count":<Integer_value>,

"hostname":<String_value>}]}
```

event_persistant_stat_info

Configuration for This table holds stat info to be persistant across reboots for event feature. resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the event stat info entries.

Operations

(click to see Properties)

- DELETE
- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/event_persistant_stat_info/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_persistant_stat_info **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_persistant_stat_info **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_persistant_stat_info resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_persistant_stat_info **? pagesize=#no;pageno=#no** Use this query-parameter to get the event_persistant_stat_info resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_persistant_stat_info **?count=yes** Use this query-parameter to get the count of event_persistant_stat_info resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_persistant_stat_info":[{

"propkey":<String_value>,

"id":<String_value>,

"propvalue":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/event_persistant_stat_info/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_persistant_stat_info":[{

"propkey":<String_value>,

"id":<String_value>,

"propvalue":<String_value>}]}
```

event_purge_setting

Configuration for Event Purge Setting resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the events.
purge_interval	<Integer>	Read-write	purge interval for instance events.

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_purge_setting **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_purge_setting **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_purge_setting resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_purge_setting **? pagesize=#no;pageno=#no** Use this query-parameter to get the event_purge_setting resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_purge_setting **?count=yes** Use this query-parameter to get the count of event_purge_setting resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_purge_setting":[{

"id":<String_value>,

"purge_interval":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/event_purge_setting/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_purge_setting":[{

"id":<String_value>,

"purge_interval":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/event_purge_setting/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"event_purge_setting":{

<b>"id":<String_value></b>,

"purge_interval":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_purge_setting":[{

"id":<String_value>,

"purge_interval":<Integer_value>}]}
```

event_resrc_threshold_violation_report

Configuration for Event Mgmt Resrc Threshold Violation Report resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
category	<String>	Read-write	category. Maximum length = 255
ip_address	<String>	Read-write	IP Address of the device. Minimum length = 1 Maximum length = 64
total_count	<Integer>	Read-write	Total event count for this device in given sampled timeframe..
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_resrc_threshold_violation_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_resrc_threshold_violation_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_resrc_threshold_violation_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_resrc_threshold_violation_report **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_resrc_threshold_violation_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_resrc_threshold_violation_report **?count=yes** Use this query-parameter to get the count of event_resrc_threshold_violation_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_resrc_threshold_violation_report":[{

"category":<String_value>,

"ip_address":<String_value>,

"total_count":<Integer_value>,

"hostname":<String_value>}]}
```

event_severity_report

Configuration for Event Mgmt Severity Based Report resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
severity	<String>	Read-write	Severity of the count of events. Maximum length = 255
total_count	<Integer>	Read-write	Total event count for this severity in given sampled timeframe..

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_severity_report **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_severity_report **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of event_severity_report resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_severity_report **? pagesize=#no;pageno=#no** Use this query-parameter to get the event_severity_report resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_severity_report **?count=yes** Use this query-parameter to get the count of event_severity_report resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_severity_report":[{

"severity":<String_value>,

"total_count":<Integer_value>}]}
```

event_summary

Configuration for Event Summary resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
Warning	<Integer>	Read-only	Total Number of Warning Events.
Minor	<Integer>	Read-only	Total number of Minor Events.
Information	<Integer>	Read-only	Total number of Information Events.
Major	<Integer>	Read-only	Total number of Major Events.
Critical	<Integer>	Read-only	Total Number of Critical Events.
category	<String>	Read-only	Category of Event.
Clear	<Integer>	Read-only	Total Number of Warning Events.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/event_summary **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/event_summary **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of

event_summary resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/event_summary **?pagesize=#no;pageno=#no** Use this query-parameter to get the event_summary resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/event_summary **?count=yes** Use this query-parameter to get the count of event_summary resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"event_summary":[{

"Warning":<Integer_value>,

"Minor":<Integer_value>,

"Information":<Integer_value>,

"Major":<Integer_value>,

"Critical":<Integer_value>,
```

```
"category":<String_value>,

"Clear":<Integer_value>}]}
```

fault_category

Configuration for Fault category details resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
device_family	<String>	Read-only	Device Family.
name	<String>	Read-only	category name.
display_name	<String>	Read-only	category display name.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/fault_category **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/fault_category **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of fault_category resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/fault_category **? pagesize=#no;pageno=#no** Use this query-parameter to get the fault_category resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/fault_category **?count=yes** Use this query-parameter to get the count of fault_category resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"fault_category":[{

"device_family":<String_value>,

"name":<String_value>,

"display_name":<String_value>}]}
```

fault_failure_object

Configuration for Fault failure object details resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the fault_failure_object.
device_family	<String>	Read-only	Device Family.
name	<String>	Read-only	failure object name.
ip_address	<String>	Read-only	Device IP Address.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/fault_failure_object **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/fault_failure_object **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of fault_failure_object resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/fault_failure_object **? pagesize=#no;pageno=#no** Use this query-parameter to get the fault_failure_object resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/fault_failure_object **?count=yes** Use this query-parameter to get the count of fault_failure_object resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"fault_failure_object":[{

"device_family":<String_value>,

"name":<String_value>,

"ip_address":<String_value>,

"id":<String_value>}]}
```

fault_severity

Configuration for Fault severity details resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-only	severity name.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/fault_severity **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/fault_severity **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of fault_severity resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/fault_severity **? pagesize=#no;pageno=#no** Use this query-parameter to get the fault_severity resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/fault_severity **?count=yes** Use this query-parameter to get the count of fault_severity resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"fault_severity":[{

"name":<String_value>}]}
```

filter_action

Configuration for event filter action properties resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_name	<String>	Read-write	Name of the parent resource.
type	<String>	Read-write	Action Type e.g. SENDMAIL,SENDSMS. Maximum length = 255
id	<String>	Read-write	Id is system generated key.
suppress_time	<Integer>	Read-write	If specified, the particular event will be suppressed for this time.
parent_id	<String>	Read-write	Id of the parent resource.
repeat_email_notification_threshold	<Integer>	Read-write	RepeatEmailNotificationThreshold - If enabled, send the

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			mail repeatedly at the specified intervals untill the event is cleared.
attachment	<String>	Read-write	Attachment for the E-mail action.
subject	<String>	Read-write	Subject of the mail. Maximum length = 255
user_message	<String>	Read-write	User Message.
profile_name	<String>	Read-write	Mail or SMS profile name.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

filter_action_props

Configuration for event filter action properties resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_name	<String>	Read-write	Name of the parent resource.
id	<String>	Read-write	Id is system generated key.
parent_id	<String>	Read-write	Id of parent resource.
value	<String>	Read-only	Property Value.
name	<String>	Read-only	Property Name.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

filter_criteria

Configuration for event filter criteria properties resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
source	<String>	Read-write	Source Ip Address of the device which sent the trap.
cmd_exec_status	<String>	Read-write	Command Execution Status if the event is generated for any config change or config save in device.
device_family	<String>	Read-write	Family of device for which we have received the event.
parent_name	<String>	Read-write	Name of the parent resource.
message	<String>	Read-write	Event Message.
parent_id	<String>	Read-write	Id of the parent resource.
cmd_auth_status	<String>	Read-write	Command Authorization Status if the event is generated for any config change or config save in device.
config_cmd	<String>	Read-write	Config Command if the event is generated for any config change in device.
id	<String>	Read-write	Id is system generated key.
category	<String>	Read-write	Category of Event.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
severity	<String>	Read-write	Severity of Event.
failureobj	<String>	Read-write	Failure objects are entity instances or counters for which an event has been generated.
category_array	<String[]>	Read-write	Category list.
failureobj_array	<String[]>	Read-write	Failure Object list.
source_array	<String[]>	Read-write	Source list.
severity_array	<String[]>	Read-write	Severity list.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

filter_schedule_details

Configuration for Filter Schedule Details resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
scheduleEndTimesEpoch	<String>	Read-write	Schedule time epoch (string representation of 11 digit numbers)..
scheduleStartTimesEpoch	<String>	Read-write	Schedule time epoch (string representation of 11 digit numbers)..
parent_name	<String>	Read-write	Name of the parent resource.
endRecurrenceTimes	<String>	Read-write	Comma separated recurrence epoch times at which job is to be executed.
id	<String>	Read-write	Id is system generated key for all the events.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
startRecurrenceTimes	<String>	Read-write	Comma separated recurrence epoch times at which job is to be executed.
parent_id	<String>	Read-write	Id of the parent resource.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

filter_scheduler

Configuration for Filter Scheduler resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
details	<filter_schedule_details[]>	Read-write	Filter Schedule Details.
recurrence_type	<String>	Read-write	Type of recurrence of schedule for the filter : daily/weekly/ specific day. Maximum length = 255
tz_offset	<Integer>	Read-write	Time zone offset..
filter_job_name	<String>	Read-write	Filter Job Name. Maximum length = 255
recurrenceOptions	<String>	Read-write	Comma separated recurrence options of job that is scheduled.
id	<String>	Read-write	Id is system generated key for all the events.
filter_status	<String>	Read-write	filter status - enabled, disabled, pause.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 255
activity_log	<String>	Read-only	Activity description.
activity_status	<String>	Read-only	Activity Status.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/filter_scheduler?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"filter_scheduler": {

"filter_job_name":<String_value>,

"id":<String_value>,

"details":[{

"scheduleStartTimesEpoch":<String_value>,

"scheduleEndTimesEpoch":<String_value>,

"parent_id":<String_value>,

"id":<String_value>,

"endRecurrenceTimes":<String_value>,
```

```
"parent_name":<String_value>,

"startRecurrenceTimes":<String_value>}],

"tz_offset":<Integer_value>,

"recurrenceOptions":<String_value>,

"filter_status":<String_value>,

"recurrence_type":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"filter_scheduler":[{

"details":[{

"scheduleEndTimesEpoch":<String_value>,

"scheduleStartTimesEpoch":<String_value>,

"parent_name":<String_value>,

"endRecurrenceTimes":<String_value>,

"id":<String_value>,

"startRecurrenceTimes":<String_value>,

"parent_id":<String_value>}],

"recurrence_type":<String_value>,

"tz_offset":<Integer_value>,
```

```
"activity_log":<String_value>,

"filter_job_name":<String_value>,

"recurrenceOptions":<String_value>,

"activity_status":<String_value>,

"id":<String_value>,

"filter_status":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/filter_scheduler/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/filter_scheduler **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/filter_scheduler **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of

filter_scheduler resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/filter_scheduler **?pagesize=#no;pageno=#no** Use this query-parameter to get the filter_scheduler resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/filter_scheduler **?count=yes** Use this query-parameter to get the count of filter_scheduler resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"filter_scheduler":[{

"details":[{

"scheduleEndTimesEpoch":<String_value>,

"scheduleStartTimesEpoch":<String_value>,

"parent_name":<String_value>,

"endRecurrenceTimes":<String_value>,
```

```
"id":<String_value>,

"startRecurrenceTimes":<String_value>,

"parent_id":<String_value>}],

"recurrence_type":<String_value>,

"tz_offset":<Integer_value>,

"activity_log":<String_value>,

"filter_job_name":<String_value>,

"recurrenceOptions":<String_value>,

"activity_status":<String_value>,

"id":<String_value>,

"filter_status":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/filter_scheduler/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"filter_scheduler":[{

"details":[{

"scheduleEndTimesEpoch":<String_value>,

"scheduleStartTimesEpoch":<String_value>,

"parent_name":<String_value>,

"endRecurrenceTimes":<String_value>,

"id":<String_value>,

"startRecurrenceTimes":<String_value>,

"parent_id":<String_value>}],

"recurrence_type":<String_value>,

"tz_offset":<Integer_value>,

"activity_log":<String_value>,

"filter_job_name":<String_value>,

"recurrenceOptions":<String_value>,

"activity_status":<String_value>,

"id":<String_value>,

"filter_status":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/filter_scheduler/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"filter_scheduler":{

<b>"id":<String_value></b>,

"filter_job_name":<String_value>,

"details":[{

"scheduleStartTimesEpoch":<String_value>,

"scheduleEndTimesEpoch":<String_value>,

"parent_id":<String_value>,

"id":<String_value>,

"endRecurrenceTimes":<String_value>,

"parent_name":<String_value>,

"startRecurrenceTimes":<String_value>}],

"tz_offset":<Integer_value>,

"recurrenceOptions":<String_value>,

"filter_status":<String_value>,

"recurrence_type":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"filter_scheduler":[{

"details":[{

"scheduleEndTimesEpoch":<String_value>,

"scheduleStartTimesEpoch":<String_value>,

"parent_name":<String_value>,

"endRecurrenceTimes":<String_value>,

"id":<String_value>,

"startRecurrenceTimes":<String_value>,

"parent_id":<String_value>}],

"recurrence_type":<String_value>,

"tz_offset":<Integer_value>,

"activity_log":<String_value>,

"filter_job_name":<String_value>,

"recurrenceOptions":<String_value>,

"activity_status":<String_value>,

"id":<String_value>,

"filter_status":<String_value>}]}
```

ns_event

Configuration for Citrix ADC Events resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the events.
counter_threshold_value	<String>	Read-only	device threshold value for any threadhold violated traps.
source	<String>	Read-only	Source from where the event is originated.
cmd_exec_status	<String>	Read-only	Command Execution Status if the event is generated for any config change or config save in device.
device_entity_type	<String>	Read-only	Device Entity Type.
entity	<String>	Read-only	Entity of Event.
device_type	<String>	Read-only	Type of device from which we have received the event.
timestamp	<Integer>	Read-only	Time when the event occured.
operation_type	<String>	Read-only	Operation Type.
category	<String>	Read-only	Category of Event.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
severity	<String>	Read-only	Severity of Event.
failureobj	<String>	Read-only	Failure objects are entity instances or counters for which an event has been generated, for example entity name.
device_family	<String>	Read-only	Family of device from which we have received the event.
rpt_sample_time	<Double>	Read-only	Report Sample time..
message	<String>	Read-only	Event Message - All the other properties recevied in event trap which do not specific fields in this resource(In key : value pair).
user_name	<String>	Read-only	Username.
config_cmd	<String>	Read-only	Config Command if the event is generated for any config change in device.
cmd_auth_status	<String>	Read-only	Command Authorization Status if the event is generated for any config change or config save in device.
device_entity_name	<String>	Read-only	Device Entity Name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
counter_actual_value	<String>	Read-only	device actual value for any threadhold violated traps.
current_value	<String>	Read-only	Current Value, cab be used while sending traps.
threshold_value	<String>	Read-only	Threshold Value, cab be used while sending traps.
entity_type	<String>	Read-only	Entity Type.
hostname	<String>	Read-only	Assign hostname to managed device, if this is not provided, name will be set as host name .
managed_device_id	<String>	Read-only	Managed Device ID.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_event **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_event **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_event resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_event **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_event resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_event **?count=yes** Use this query-parameter to get the count of ns_event resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_event":[{

"counter_threshold_value":<String_value>,

"source":<String_value>,

"cmd_exec_status":<String_value>,

"device_entity_type":<String_value>,

"entity":<String_value>,

"device_type":<String_value>,

"timestamp":<Integer_value>,

"operation_type":<String_value>,

"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"failureobj":<String_value>,

"device_family":<String_value>,
```

```
"rpt_sample_time":<Double_value>,

"message":<String_value>,

"user_name":<String_value>,

"config_cmd":<String_value>,

"cmd_auth_status":<String_value>,

"device_entity_name":<String_value>,

"counter_actual_value":<String_value>,

"current_value":<String_value>,

"threshold_value":<String_value>,

"entity_type":<String_value>,

"hostname":<String_value>,

"matched_filters":<String_value>,

"managed_device_id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_event/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_event":[{

"counter_threshold_value":<String_value>,

"source":<String_value>,

"cmd_exec_status":<String_value>,

"device_entity_type":<String_value>,

"entity":<String_value>,

"device_type":<String_value>,

"timestamp":<Integer_value>,

"operation_type":<String_value>,

"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"failureobj":<String_value>,

"device_family":<String_value>,

"rpt_sample_time":<Double_value>,

"message":<String_value>,
```

```
"user_name":<String_value>,

"config_cmd":<String_value>,

"cmd_auth_status":<String_value>,

"device_entity_name":<String_value>,

"counter_actual_value":<String_value>,

"current_value":<String_value>,

"threshold_value":<String_value>,

"entity_type":<String_value>,

"hostname":<String_value>,

"matched_filters":<String_value>,

"managed_device_id":<String_value>}]}
```


ns_event_archive

Configuration for Citrix ADC Events resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the events.
counter_threshold_value	<String>	Read-only	device threshold value for any threadhold violated traps.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
source	<String>	Read-only	Source from where the event is originated.
cmd_exec_status	<String>	Read-only	Command Execution Status if the event is generated for any config change or config save in device.
device_entity_type	<String>	Read-only	Device Entity Type.
entity	<String>	Read-only	Entity of Event.
device_type	<String>	Read-only	Type of device from which we have received the event.
timestamp	<Integer>	Read-only	Time when the event occured.
operation_type	<String>	Read-only	Operation Type.
category	<String>	Read-only	Category of Event.
severity	<String>	Read-only	Severity of Event.
failureobj	<String>	Read-only	Failure objects are entity instances or counters for which an event has been generated, for example entity name.
device_family	<String>	Read-only	Family of device from which we have received the event.
rpt_sample_time	<Double>	Read-only	Report Sample time..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
message	<String>	Read-only	Event Message - All the other properties recevied in event trap which do not specific fields in this resource(In key : value pair).
user_name	<String>	Read-only	Username.
config_cmd	<String>	Read-only	Config Command if the event is generated for any config change in device.
cmd_auth_status	<String>	Read-only	Command Authorization Status if the event is generated for any config change or config save in device.
device_entity_name	<String>	Read-only	Device Entity Name.
counter_actual_value	<String>	Read-only	device actual value for any threadhold violated traps.
current_value	<String>	Read-only	Current Value, cab be used while sending traps.
threshold_value	<String>	Read-only	Threshold Value, cab be used while sending traps.
entity_type	<String>	Read-only	Entity Type.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hostname	<String>	Read-only	Assign hostname to managed device, if this is not provided, name will be set as host name .
managed_device_id	<String>	Read-only	Managed Device ID.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_event_archive **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_event_archive **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_event_archive resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_event_archive **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_event_archive resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_event_archive **?count=yes** Use this query-parameter to get the count of ns_event_archive resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_event_archive":[{

"counter_threshold_value":<String_value>,

"source":<String_value>,

"cmd_exec_status":<String_value>,

"device_entity_type":<String_value>,

"entity":<String_value>,

"device_type":<String_value>,

"timestamp":<Integer_value>,

"operation_type":<String_value>,

"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"failureobj":<String_value>,

"device_family":<String_value>,

"rpt_sample_time":<Double_value>,

"message":<String_value>,

"user_name":<String_value>,

"config_cmd":<String_value>,

"cmd_auth_status":<String_value>,

"device_entity_name":<String_value>,
```

```
"counter_actual_value":<String_value>,

"current_value":<String_value>,

"threshold_value":<String_value>,

"entity_type":<String_value>,

"hostname":<String_value>,

"matched_filters":<String_value>,

"managed_device_id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_event_archive/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_event_archive":[{

"counter_threshold_value":<String_value>,
```

```
"source":<String_value>,

"cmd_exec_status":<String_value>,

"device_entity_type":<String_value>,

"entity":<String_value>,

"device_type":<String_value>,

"timestamp":<Integer_value>,

"operation_type":<String_value>,

"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"failureobj":<String_value>,

"device_family":<String_value>,

"rpt_sample_time":<Double_value>,

"message":<String_value>,

"user_name":<String_value>,

"config_cmd":<String_value>,

"cmd_auth_status":<String_value>,

"device_entity_name":<String_value>,

"counter_actual_value":<String_value>,

"current_value":<String_value>,

"threshold_value":<String_value>,

"entity_type":<String_value>,
```

```
"hostname":<String_value>,

"matched_filters":<String_value>,

"managed_device_id":<String_value>}]}
```

ns_event_message

Configuration for Citrix ADC Events resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
entity	<String>	Read-write	Entity of Event. Minimum length = 1 Maximum length = 1024
id	<String>	Read-write	Id is system generated key for all the events.
counter_threshold_value	<String>	Read-only	device threshold value for any threadhold violated traps.
source	<String>	Read-only	Source.
cmd_exec_status	<String>	Read-only	Command Execution Status if the event is generated for any config change or config save in device.
device_entity_type	<String>	Read-only	Device Entity Type.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
device_type	<String>	Read-only	Device Type.
timestamp	<Integer>	Read-only	Time when the event occured.
operation_type	<String>	Read-only	Operation Type.
category	<String>	Read-only	Category of Event.
severity	<String>	Read-only	Severity of Event.
failureobj	<String>	Read-only	Failure objects are entity instances or counters for which an event has been generated, for example entity name.
source_event_id	<Integer>	Read-only	Internal Event ID used in the source device..
history	<String>	Read-only	History of the Event with same entity.
device_family	<String>	Read-only	Device Family.
total_count	<Integer>	Read-only	total_count.
message	<String>	Read-only	Event Message - All the other properties recevied in event trap which do not specific fields in this resource(In key : value pair).
user_name	<String>	Read-only	Username.
device_entity_name	<String>	Read-only	Device Entity Name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
config_cmd	<String>	Read-only	Config Command if the event is generated for any config change in device.
cmd_auth_status	<String>	Read-only	Command Authorization Status if the event is generated for any config change or config save in device.
counter_actual_value	<String>	Read-only	device actual value for any threadhold violated traps.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_event_message **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_event_message **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_event_message resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_event_message **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_event_message resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_event_message **?count=yes** Use this query-parameter to get the count of ns_event_message resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_event_message":[{

"counter_threshold_value":<String_value>,

"source":<String_value>,

"cmd_exec_status":<String_value>,

"device_entity_type":<String_value>,

"trap_id":<Integer_value>,

"entity":<String_value>,

"device_type":<String_value>,

"timestamp":<Integer_value>,

"operation_type":<String_value>,

"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"failureobj":<String_value>,

"source_event_id":<Integer_value>,
```

```
"history":<String_value>,

"device_family":<String_value>,

"total_count":<Integer_value>,

"message":<String_value>,

"user_name":<String_value>,

"device_entity_name":<String_value>,

"config_cmd":<String_value>,

"cmd_auth_status":<String_value>,

"counter_actual_value":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_event_message/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_event_message":[{

"counter_threshold_value":<String_value>,

"source":<String_value>,

"cmd_exec_status":<String_value>,

"device_entity_type":<String_value>,

"trap_id":<Integer_value>,

"entity":<String_value>,

"device_type":<String_value>,

"timestamp":<Integer_value>,

"operation_type":<String_value>,

"id":<String_value>,

"category":<String_value>,

"severity":<String_value>,

"failureobj":<String_value>,

"source_event_id":<Integer_value>,

"history":<String_value>,

"device_family":<String_value>,

"total_count":<Integer_value>,

"message":<String_value>,

"user_name":<String_value>,

"device_entity_name":<String_value>,
```

```
"config_cmd":<String_value>,

"cmd_auth_status":<String_value>,

"counter_actual_value":<String_value>}]}
```

ns_event_summary

Configuration for Citrix ADC event devices summary resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
longitude	<Double>	Read-write	longitude.
is_agent	<Boolean>	Read-write	Is this device acting as agent..
latitude	<Double>	Read-write	latitude.
ip_address	<String>	Read-write	ip_address of the device..
cityname	<String>	Read-write	City Name. Maximum length = 255
total_critical_events	<Integer>	Read-only	Total Critical events city wise..
tenant_id	<String>	Read-only	Tenant Id.
total_major_events	<Integer>	Read-only	Total major events..
instance_state	<String>	Read-only	State of device, UP only if device accessible.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ns_event_summary **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_event_summary **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_event_summary resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_event_summary **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_event_summary resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_event_summary **?count=yes** Use this query-parameter to get the count of ns_event_summary resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_event_summary":[{

"total_critical_events":<Integer_value>,

"longitude":<Double_value>,
```

```
"tenant_id":<String_value>,

"is_agent":<Boolean_value>,

"latitude":<Double_value>,

"total_major_events":<Integer_value>,

"ip_address":<String_value>,

"instance_state":<String_value>,

"cityname":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ns_event_summary/ip_address_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_event_summary":[{
```

```
"total_critical_events":<Integer_value>,

"longitude":<Double_value>,

"tenant_id":<String_value>,

"is_agent":<Boolean_value>,

"latitude":<Double_value>,

"total_major_events":<Integer_value>,

"ip_address":<String_value>,

"instance_state":<String_value>,

"cityname":<String_value>}]}
```

poll_activity_status

Configuration for Activity Status resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
instance_level_progress	<Integer>	Read-write	Instance level Polling Progress Percentage. Maximum value =
status	<String>	Read-write	Status.
message	<String>	Read-write	Message.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
total_instances	<Integer>	Read-write	Total number of devices to be polled.
timestamp	<Double>	Read-write	timestamp in seconds.
id	<String>	Read-write	Id is system generated key for all the task logs.
act_id	<String>	Read-only	Activity Id.
module	<String>	Read-only	Module name in which the Polling is done.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/poll_activity_status **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/poll_activity_status **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of poll_activity_status resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/poll_activity_status **? pagesize=#no;pageno=#no** Use this query-parameter to get the poll_activity_status resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/poll_activity_status **?count=yes** Use this query-parameter to get the count of poll_activity_status resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"poll_activity_status":[{

"instance_level_progress":<Integer_value>,

"status":<String_value>,

"message":<String_value>,

"ns_ip_address":<String_value>,

"total_instances":<Integer_value>,

"act_id":<String_value>,

"timestamp":<Double_value>,

"id":<String_value>,

"module":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/poll_activity_status/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"poll_activity_status":[{

"instance_level_progress":<Integer_value>,

"status":<String_value>,

"message":<String_value>,

"ns_ip_address":<String_value>,

"total_instances":<Integer_value>,

"act_id":<String_value>,

"timestamp":<Double_value>,

"id":<String_value>,

"module":<String_value>}]}
```

syslog_filter

Configuration for Syslog Suppress Filters resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_enabled	<Boolean>	Read-write	Filter is enabled or disabled.
name	<String>	Read-write	Filter Name. Maximum length = 255
id	<String>	Read-write	Id is system generated key.
severity	<String>	Read-write	Comma separated string of severities to be suppressed.
message	<String>	Read-write	Wildcard substring for syslog message to be suppressed.
facilities	<String>	Read-write	Comma separated string of facilities to be suppressed.
devices	<String>	Read-write	Comma separated string of devices to be suppressed.
facilities_array	<String[]>	Read-write	Facilities list.
devices_array	<String[]>	Read-write	Devices list.
severity_array	<String[]>	Read-write	Severity list.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET

- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/syslog_filter?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"syslog_filter": {

"devices_array":<String_value[]>,

"id":<String_value>,

"is_enabled":<Boolean_value>,

"name":<String_value>,

"devices":<String_value>,

"facilities_array":<String_value[]>,

"severity":<String_value>,

"message":<String_value>,

"severity_array":<String_value[]>,

"facilities":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_filter":[{

"is_enabled":<Boolean_value>,
```

```
"name":<String_value>,

"id":<String_value>,

"severity":<String_value>,

"message":<String_value>,

"facilities":<String_value>,

"devices":<String_value>,

"facilities_array":<String_value>,

"devices_array":<String_value>,

"severity_array":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/syslog_filter/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/syslog_filter **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/syslog_filter **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of syslog_filter resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/syslog_filter **?pagesize=#no;pageno=#no** Use this query-parameter to get the syslog_filter resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/syslog_filter **?count=yes** Use this query-parameter to get the count of syslog_filter resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_filter":[{

"is_enabled":<Boolean_value>,

"name":<String_value>,

"id":<String_value>,

"severity":<String_value>,
```

```
"message":<String_value>,

"facilities":<String_value>,

"devices":<String_value>,

"facilities_array":<String_value>,

"devices_array":<String_value>,

"severity_array":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/syslog_filter/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_filter":[{

"is_enabled":<Boolean_value>,

"name":<String_value>,
```

```
"id":<String_value>,

"severity":<String_value>,

"message":<String_value>,

"facilities":<String_value>,

"devices":<String_value>,

"facilities_array":<String_value>,

"devices_array":<String_value>,

"severity_array":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/syslog_filter/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"syslog_filter":{

<b>"id":<String_value></b>,

"devices_array":<String_value[]>,

"is_enabled":<Boolean_value>,

"name":<String_value>,
```

```
"devices":<String_value>,

"facilities_array":<String_value[]>,

"severity":<String_value>,

"message":<String_value>,

"severity_array":<String_value[]>,

"facilities":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_filter":[{

"is_enabled":<Boolean_value>,

"name":<String_value>,

"id":<String_value>,

"severity":<String_value>,

"message":<String_value>,

"facilities":<String_value>,

"devices":<String_value>,

"facilities_array":<String_value>,

"devices_array":<String_value>,

"severity_array":<String_value>}]}
```

syslog_generic

Configuration for Syslog generic table resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the syslog generic record..
source	<String>	Read-only	source from where the syslog is generated.
device_family	<String>	Read-only	Device family.
event_id	<Double>	Read-only	;ltTODO;gt.
system_name	<String>	Read-only	Device System Name.
full_message	<String>	Read-only	Syslog original Message..
isparsed	<Boolean>	Read-only	Whethersyslog message is parsed successfully or not.
ttime	<Double>	Read-only	ttime when the syslog is generated.
device_type	<String>	Read-only	Device Type.
host_name	<String>	Read-only	Assign hostname to managed device, if this is not provided,

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			name will be set as host name .
type	<String>	Read-only	Type Name..
syslog_msg	<String>	Read-only	Syslog original Message..
severity	<String>	Read-only	severity.
module	<String>	Read-only	Module Name..

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/syslog_generic **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/syslog_generic **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of syslog_generic resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/syslog_generic **? pagesize=#no;pageno=#no** Use this query-parameter to get the syslog_generic resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/syslog_generic **?count=yes** Use this query-parameter to get the count of syslog_generic resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_generic":[{

"source":<String_value>,

"device_family":<String_value>,

"event_id":<Double_value>,

"system_name":<String_value>,

"full_message":<String_value>,

"isparsed":<Boolean_value>,

"ttime":<Double_value>,

"device_type":<String_value>,

"host_name":<String_value>,

"id":<String_value>,

"type":<String_value>,

"syslog_msg":<String_value>,

"severity":<String_value>,

"module":<String_value>}]}
```

syslog_severity

Configuration for Syslog severity details resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-only	severity name.
code	<String>	Read-only	severity code.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/syslog_severity **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/syslog_severity **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of syslog_severity resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/syslog_severity **? pagesize=#no;pageno=#no** Use this query-parameter to get the syslog_severity resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/syslog_severity **?count=yes** Use this query-parameter to get the count of syslog_severity resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_severity":[{
```

```
"name":<String_value>,

"code":<String_value>}]}
```

trap_settings

Configuration for trap severities resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
source	<String>	Read-write	Source.
default_severity	<String>	Read-write	default severity of trap.
device_family	<String>	Read-write	Device Family.
oid	<String>	Read-write	Trap OID.
user_defined_severity	<String>	Read-write	user defined severity of trap.
trap_description	<String>	Read-write	Trap Description.
trap_category	<String>	Read-write	Trap Category.
is_suppress	<Boolean>	Read-write	if trap needs to be suppressed or not.
id	<String>	Read-write	Id is system generated key.
source_array	<String[]>	Read-write	Source list.

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/trap_settings **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/trap_settings **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of trap_settings resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/trap_settings **?pagesize=#no;pageno=#no** Use this query-parameter to get the trap_settings resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/trap_settings **?count=yes** Use this query-parameter to get the count of trap_settings resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"trap_settings":[{

"source":<String_value>,

"default_severity":<String_value>,

"device_family":<String_value>,

"oid":<String_value>,
```

```
"user_defined_severity":<String_value>,

"trap_description":<String_value>,

"trap_category":<String_value>,

"is_suppress":<Boolean_value>,

"id":<String_value>,

"source_array":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/trap_settings/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"trap_settings":[{

"source":<String_value>,

"default_severity":<String_value>,
```

```
"device_family":<String_value>,

"oid":<String_value>,

"user_defined_severity":<String_value>,

"trap_description":<String_value>,

"trap_category":<String_value>,

"is_suppress":<Boolean_value>,

"id":<String_value>,

"source_array":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/trap_settings/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"trap_settings":{

<b>"id":<String_value></b>,

"trap_description":<String_value>,

"is_suppress":<Boolean_value>,

"device_family":<String_value>,
```

```
"source_array":<String_value[]>,

"trap_category":<String_value>,

"source":<String_value>,

"default_severity":<String_value>,

"oid":<String_value>,

"user_defined_severity":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"trap_settings":[{

"source":<String_value>,

"default_severity":<String_value>,

"device_family":<String_value>,

"oid":<String_value>,

"user_defined_severity":<String_value>,

"trap_description":<String_value>,

"trap_category":<String_value>,

"is_suppress":<Boolean_value>,

"id":<String_value>,

"source_array":<String_value>}]}
```

Instances

- Citrix-ADC-VPX
- Citrix-SD-WAN
- Citrix-SD-WAN-Accelerator
- Common-Resources
- enable_ipv6_traffic

enable_ipv6_traffic

Configuration for Enable IPv6 Traffic resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key.

Operations

(click to see Properties)

- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get

URL: https://<MGMT-IP>/nitro/v2/config/enable_ipv6_traffic **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"enable_ipv6_traffic":[{

"id":<String_value>,

"is_ipv6_enabled":<Boolean_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/enable_ipv6_traffic/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"enable_ipv6_traffic":{

"id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"enable_ipv6_traffic":[{
```

```
"id":<String_value>,

"is_ipv6_enabled":<Boolean_value>}]}
```

Citrix-ADC-VPX

- adc_params
- adc_update_appflow_info
- adc_vserver_appflow_config
- adc_vserver_appflow_info
- admin_ns_config
- blx_device_profile
- ns
- ns_appflow_param_config
- ns_callhome
- ns_clag_interface
- ns_clag_node
- ns_cluster
- ns_cluster_nodegroup
- ns_clusternode
- ns_command
- ns_config_replicate
- ns_detail_vlan
- ns_device_profile
- ns_emon_poll_policy
- ns_filterpolicy_template
- ns_hapair_template
- ns_image
- ns_interface
- ns_l7_latency_config

- ns_mpxcluster_template
- ns_network_interface
- ns_ns_appflowcollector
- ns_ns_appflowparam
- ns_ns_ip
- ns_ns_ipv6
- ns_ns_mode
- ns_ns_rollbackcmd
- ns_ns_runningconfig
- ns_ns_savedconfig
- ns_ns_timeout
- ns_partition
- ns_save_config
- ns_snmp_config
- ns_sslcipher_binding
- ns_sslparameter
- ns_sslprofile
- ns_sslprofile_cipher_binding
- ns_sslvserver_cipher_binding
- ns_syslog_config
- ns_syslog_task
- ns_upgrade
- ns_version
- ns_vlan
- ns_vm_image
- ns_vmacs
- ns_vmacs_free_count
- ns_vserver_appflow_config

adc_params

Configuration for Citrix ADC Info Parameters resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
httpquerywithurl	<String>	Read-write	URL query param flag.. Minimum length = 1 Maximum length = 10
template_interval	<Integer>	Read-write	Template refresh interval.
httpxforwardedfor	<String>	Read-write	HTTP x-Forwarded for header flag.. Minimum length = 1 Maximum length = 10
gateway_deployment	<Boolean>	Read-write	Is this device acting as Gateway..
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
geo_support	<Boolean>	Read-write	Is this device configured to support GEO location..
httpcookie	<String>	Read-write	Cookie-Header flag.. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 10
id	<String>	Read-write	Netscaler ID.. Minimum length = 1

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/adc_params?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"adc_params": {

"ns_ip_address":<String_value>,

"gateway_deployment":<Boolean_value>,

"id":<String_value>,

"template_interval":<Integer_value>,

"httpquerywithurl":<String_value>,

"geo_support":<Boolean_value>,

"httpxforwardedfor":<String_value>,

"httpcookie":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"adc_params":[{

"httpquerywithurl":<String_value>,

"template_interval":<Integer_value>,

"httpxforwardedfor":<String_value>,

"gateway_deployment":<Boolean_value>,

"ns_ip_address":<String_value>,

"geo_support":<Boolean_value>,

"httpcookie":<String_value>,

"id":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/adc_params/ **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/adc_params **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/adc_params **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of adc_params resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/adc_params **?pagesize=#no;pageno=#no** Use this query-parameter to get the adc_params resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/adc_params **?count=yes** Use this query-parameter to get the count of adc_params resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"adc_params":[{

"httpquerywithurl":<String_value>,

"template_interval":<Integer_value>,

"httpxforwardedfor":<String_value>,

"gateway_deployment":<Boolean_value>,

"ns_ip_address":<String_value>,

"geo_support":<Boolean_value>,

"httpcookie":<String_value>,

"id":<String_value>}]}
```

MODIFY

URL: https://<MGMT-IP>/nitro/v2/config/adc_params/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"adc_params":{

"ns_ip_address":<String_value>,

"gateway_deployment":<Boolean_value>,

"id":<String_value>,
```

```
"template_interval":<Integer_value>,

"httpquerywithurl":<String_value>,

"geo_support":<Boolean_value>,

"httpxforwardedfor":<String_value>,

"httpcookie":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"adc_params":[{

"httpquerywithurl":<String_value>,

"template_interval":<Integer_value>,

"httpxforwardedfor":<String_value>,

"gateway_deployment":<Boolean_value>,

"ns_ip_address":<String_value>,

"geo_support":<Boolean_value>,

"httpcookie":<String_value>,

"id":<String_value>}]}
```

adc_update_appflow_info

Configuration for Execute the Appflow Info Query through Remote Proxy Handler resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
transactionlog	<String>	Read-write	transactionlog.
priority	<String>	Read-write	priority.
transport	<String>	Read-write	Transport type.. Minimum length = 1 Maximum length = 20
policy_name	<String>	Read-write	policy_name. Minimum length = 1 Maximum length = 256
vsvr_name	<String>	Read-write	Vserver name. Minimum length = 1 Maximum length = 100
webinsight	<String>	Read-write	webinsight.
videoanalytics	<String>	Read-write	videoanalytics.
rule	<String>	Read-write	Rule.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
agent_list	<String>	Read-write	Agent List, on which traffic will flow.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
appflowlog_effective	<String>	Read-write	Appflowlog effective (validating AppFlow config) enabled or disabled.
securityinsight	<String>	Read-write	securityinsight.
icalog	<String>	Read-write	ICA log.
collector_ip	<String>	Read-write	Collector IP Address. Minimum length = 1 Maximum length = 64
export_option	<String>	Read-write	Export Options: TCP, ICA.
appflow_policy_rule	<String>	Read-write	Appflow policy rule.
es4nslog	<String>	Read-write	ESNS enable.
collectors	<String>	Read-write	collectors.
action	<String>	Read-write	action.
type	<String>	Read-write	type.
clientsidemeasurements	<String>	Read-write	clientsidemeasurements.
metricslog	<Boolean>	Read-write	metricslog.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/adc_update_appflow_info **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/adc_update_appflow_info **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of adc_update_appflow_info resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/adc_update_appflow_info **?pagesize=#no;pageno=#no** Use this query-parameter to get the adc_update_appflow_info resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/adc_update_appflow_info **?count=yes** Use this query-parameter to get the count of adc_update_appflow_info resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"adc_update_appflow_info":[{

"transactionlog":<String_value>,

"priority":<String_value>,

"transport":<String_value>,

"policy_name":<String_value>,

"vsvr_name":<String_value>,

"webinsight":<String_value>,
```

```
"videoanalytics":<String_value>,

"rule":<String_value>,

"ns_ip_address":<String_value>,

"agent_list":<String_value>,

"appflowlog_effective":<String_value>,

"securityinsight":<String_value>,

"icalog":<String_value>,

"collector_ip":<String_value>,

"export_option":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"collectors":<String_value>,

"action":<String_value>,

"type":<String_value>,

"clientsidemeasurements":<String_value>,

"metricslog":<Boolean_value>}]}
```

adc_vserver_appflow_config

Configuration for Virtual server appflow config on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
adc_params_info	<adc_params[]>	Read-write	ADC params info object.
adc_appflow_info	<adc_vserver_appflow_info[]>	Read-write	ADC appflow info object.
tenant_id	<String>	Read-write	TenantID.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_config?onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"adc_vserver_appflow_config": {

"adc_params_info":[{

"ns_ip_address":<String_value>,

"gateway_deployment":<Boolean_value>,

"id":<String_value>,

"httpquerywithurl":<String_value>,

"template_interval":<Integer_value>,

"geo_support":<Boolean_value>,
```

```
"httpxforwardedfor":<String_value>,

"httpcookie":<String_value>}],

"tenant_id":<String_value>,

"adc_appflow_info":[{

"ns_ip_address":<String_value>,

"export_option":<String_value[]>,

"name":<String_value>,

"type":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>}]}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"adc_vserver_appflow_config":[{

"adc_params_info":[{

"httpquerywithurl":<String_value>

"template_interval":<Integer_value>

"httpxforwardedfor":<String_value>

"gateway_deployment":<Boolean_value>
```

```
"ns_ip_address":<String_value>

"geo_support":<Boolean_value>

"httpcookie":<String_value>

"id":<String_value>}],

"adc_appflow_info":[{

"export_option":<String_value>

"partition_name":<String_value>

"transport_mode":<String_value>

"appflow_policy_rule":<String_value>

"es4nslog":<String_value>

"name":<String_value>

"type":<String_value>

"ns_ip_address":<String_value>}],

"tenant_id":<String_value>,

"act_id":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_config/ **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_config **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_config **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of adc_vserver_appflow_config resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_config **?pagesize=#no;pageno=#no** Use this query-parameter to get the adc_vserver_appflow_config resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_config **?count=yes** Use this query-parameter to get the count of adc_vserver_appflow_config resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"adc_vserver_appflow_config":[{

"adc_params_info":[{

"httpquerywithurl":<String_value>

"template_interval":<Integer_value>

"httpxforwardedfor":<String_value>

"gateway_deployment":<Boolean_value>

"ns_ip_address":<String_value>

"geo_support":<Boolean_value>

"httpcookie":<String_value>

"id":<String_value>}],

"adc_appflow_info":[{

"export_option":<String_value>

"partition_name":<String_value>

"transport_mode":<String_value>

"appflow_policy_rule":<String_value>

"es4nslog":<String_value>

"name":<String_value>

"type":<String_value>

"ns_ip_address":<String_value>}],

"tenant_id":<String_value>,
```

```
"act_id":<String_value>}]}
```

MODIFY

URL: https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_config/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"adc_vserver_appflow_config":{

"adc_params_info":[{

"ns_ip_address":<String_value>,

"gateway_deployment":<Boolean_value>,

"id":<String_value>,

"httpquerywithurl":<String_value>,

"template_interval":<Integer_value>,

"geo_support":<Boolean_value>,

"httpxforwardedfor":<String_value>,

"httpcookie":<String_value>}],

"tenant_id":<String_value>,

"adc_appflow_info":[{

"ns_ip_address":<String_value>,
```

```
"export_option":<String_value[]>,

"name":<String_value>,

"type":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>}]}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"adc_vserver_appflow_config":[{

"adc_params_info":[{

"httpquerywithurl":<String_value>

"template_interval":<Integer_value>

"httpxforwardedfor":<String_value>

"gateway_deployment":<Boolean_value>

"ns_ip_address":<String_value>

"geo_support":<Boolean_value>

"httpcookie":<String_value>

"id":<String_value>}],

"adc_appflow_info":[{
```

```
"export_option":<String_value>

"partition_name":<String_value>

"transport_mode":<String_value>

"appflow_policy_rule":<String_value>

"es4nslog":<String_value>

"name":<String_value>

"type":<String_value>

"ns_ip_address":<String_value>}],

"tenant_id":<String_value>,

"act_id":<String_value>}]}
```

adc_vserver_appflow_info

Configuration for Virtual server appflow info on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
export_option	<String[]>	Read-write	Export Options: TCP, ICA.
transport_mode	<String>	Read-write	Transport Mode [Logstream/ Appflow].
appflow_policy_rule	<String>	Read-write	Appflow policy rule.
es4nslog	<String>	Read-write	ESNS enable.
name	<String>	Read-write	Virtual server name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 255
type	<String>	Read-write	Tells whether virtual server type. Minimum length = 2 Maximum length = 64
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_info?onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"adc_vserver_appflow_info": {

"ns_ip_address":<String_value>,
```

```
"name":<String_value>,

"type":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"export_option":<String_value[]>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"adc_vserver_appflow_info":[{

"export_option":<String_value>,

"partition_name":<String_value>,

"transport_mode":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"name":<String_value>,

"type":<String_value>,

"ns_ip_address":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_info/ **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_info **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_info **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of adc_vserver_appflow_info resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_info **? pagesize=#no;pageno=#no** Use this query-parameter to get the adc_vserver_appflow_info resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_info **?count=yes** Use this query-parameter to get the count of adc_vserver_appflow_info resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"adc_vserver_appflow_info":[{

"export_option":<String_value>,

"partition_name":<String_value>,

"transport_mode":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"name":<String_value>,

"type":<String_value>,

"ns_ip_address":<String_value>}]}
```

MODIFY

URL: https://<MGMT-IP>/nitro/v2/config/adc_vserver_appflow_info/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"adc_vserver_appflow_info":{

"ns_ip_address":<String_value>,

"name":<String_value>,

"type":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"export_option":<String_value[]>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"adc_vserver_appflow_info":[{

"export_option":<String_value>,

"partition_name":<String_value>,

"transport_mode":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"name":<String_value>,

"type":<String_value>,

"ns_ip_address":<String_value>}]}
```

admin_ns_config

Configuration for Admin configuration on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

MODIFY

URL: https://<MGMT-IP>/nitro/v2/config/admin_ns_config/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"admin_ns_config":{

"ns_ip_address":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"admin_ns_config":[{

"ns_ip_address":<String_value>}]}
```

blx_device_profile

Configuration for Device profile for Citrix ADC BLX instances resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Profile Name. Minimum length = 1 Maximum length = 128
svm_ns_comm	<String>	Read-write	Communication protocol (http or https) with Instances. Minimum length = 1 Maximum length = 10
use_global_setting_for_communication_with_ns	<Boolean>	Read-write	True, if the communication with Instance needs to be global and not device specific.
id	<String>	Read-write	Id is system generated key for all the device profiles.
type	<String>	Read-write	Profile Type, This must be with in specified supported

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			instance types: blx,sdvanvw,ns,nssdx,cbwanopt,cpx. Minimum length = 1 Maximum length = 128
ns_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix ADC profiles.
password	<String>	Read-write	Instance credentials.Password for this profile. Minimum length = 1 Maximum length = 127
snmpsecuritylevel	<String>	Read-write	SNMP v3 security level for this profile.
username	<String>	Read-write	Instance credentials.Username provided in the profile will be used to contact the instance. Minimum length = 1 Maximum length = 127
snmpauthprotocol	<String>	Read-write	SNMP v3 auth protocol for this profile.
ssh_port	<String>	Read-write	SSH port to connect to the device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
snmpprivprotocol	<String>	Read-write	SNMP v3 priv protocol for this profile.
host_password	<String>	Read-write	Host Password for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpversion	<String>	Read-write	SNMP version for this profile.
passphrase	<String>	Read-write	Passphrase with which private key is encrypted.
host_username	<String>	Read-write	Host User Name for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpsecurityname	<String>	Read-write	SNMP v3 security name for this profile. Maximum length = 31
ssl_private_key	<String>	Read-write	SSL Private Key for key based authentication.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ssl_cert	<String>	Read-write	SSL Certificate for certificate based authentication.
http_port	<Integer>	Read-write	HTTP port to connect to the device.
snmpcommunity	<String>	Read-write	SNMP community for this profile. Maximum length = 31
https_port	<Integer>	Read-write	HTTPS port to connect to the device.
max_wait_time_reboot	<String>	Read-write	Max waiting time to reboot Citrix ADC.
snmpprivpassword	<String>	Read-write	SNMP v3 priv password for this profile. Minimum length = 8 Maximum length = 31
cb_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix SD-WAN profiles.
snmpauthpassword	<String>	Read-write	SNMP v3 auth password for this profile. Minimum length = 8 Maximum length = 31

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_default	<Boolean>	Read-only	True, if this profile is system generated and can not be deleted.
tenant_id	<String>	Read-only	Tenant Id of device profile.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/blx_device_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"blx_device_profile": {

<b>"name":<String_value></b>,

<b>"password":<String_value></b>,

<b>"username":<String_value></b>,

"snmpsecuritylevel":<String_value>,

"svm_ns_comm":<String_value>,

"ssh_port":<String_value>,

"host_password":<String_value>,
```

```
"snmpversion":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"type":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"ns_profile_name":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"blx_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"tenant_id":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,
```

```
"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/blx_device_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/blx_device_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/blx_device_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of blx_device_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/blx_device_profile **?pagesize=#no;pageno=#no** Use this query-parameter to get the blx_device_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/blx_device_profile **?count=yes** Use this query-parameter to get the count of blx_device_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"blx_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,
```

```
"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"tenant_id":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,
```

```
"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/blx_device_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"blx_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,
```

```
"is_default":<Boolean_value>,

"type":<String_value>,

"tenant_id":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,
```

```
"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/blx_device_profile/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"blx_device_profile":{

<b>"id":<String_value></b>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"name":<String_value>,

"svm_ns_comm":<String_value>,

"ssh_port":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"type":<String_value>,

"ssl_private_key":<String_value>,
```

```
"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"ns_profile_name":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"blx_device_profile":[{

"name":<String_value>,
```

```
"svm_ns_comm":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"tenant_id":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,
```

```
"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

ns

Configuration for Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address for this managed device. Minimum length = 1 Maximum length = 128
password	<String>	Read-write	Password for specified user on Citrix ADC Instance. Minimum length = 1 Maximum length = 127
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
throughput_limit	<Double>	Read-write	Throughput Limit in Mbps set for VM Instance.
vlan_id_0_1	<Integer>	Read-write	VLAN id for the management interface 0/1. Maximum value =
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
plugin_ip_address	<String>	Read-write	Signaling IP Address. Minimum length = 1 Maximum length = 64
vlan_type	<Integer>	Read-write	VLAN Type, Citrix ADC or L2 VLAN. Maximum value =
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.
nsvlan_tagged	<Boolean>	Read-write	NSVLAN Tagged.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1 Maximum length = 64
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.
number_of_ssl_cores_up	<Integer>	Read-write	Number of SSL Cores Up.
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device. Minimum length = 1 Maximum length = 64
if_0_1	<Boolean>	Read-write	Network 0/1 on VM Instance.
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
host_ip_address	<String>	Read-write	Host IPAddress where VM is provisioned. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key for all the managed devices.
ipv6_address	<String>	Read-write	IPv6 Address.
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
number_of_acu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
is_clip	<Boolean>	Read-write	Is Clip.
device_family	<String>	Read-write	Device Family. Minimum length = 1 Maximum length = 64
type	<String>	Read-write	Type of device, (Xen | NS).

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	Assign throughput in Mbps to VM Instance.
template_name	<String>	Read-write	Template Name, This parameter is used while provisioning VM Instance with template, template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64
iscco	<Boolean>	Read-write	Is CCO.
number_of_scu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
license	<String>	Read-write	Feature License for Citrix ADC Instance, needs to be set while provisioning

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			(standard, enterprise, platinum). Minimum length = 1 Maximum length = 64
domain_name	<String>	Read-write	Domain name of VM Device. Minimum length = 1 Maximum length = 128
if_0_2	<Boolean>	Read-write	Network 0/2 on VM Instance.
image_name	<String>	Read-write	Image Name, This parameter is used while provisioning VM Instance with XVA image, template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vm_memory_total	<Double>	Read-write	Total Memory of VM Instance in MB. 2048MB, 5120MB.
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
description	<String>	Read-write	Description of managed device. Minimum length = 1 Maximum length = 512
username	<String>	Read-write	User Name (except nsroot) to be configured on Citrix ADC Instance. Minimum length = 1 Maximum length = 127
fips_partition_name	<String>	Read-write	FIPS Partition Name. Minimum length = 1 Maximum length = 128
nsvlan_id	<Integer>	Read-write	VLAN Id. Maximum value =
num_pes	<Integer>	Read-write	Total number of PEs.
burst_priority	<Integer>	Read-write	Burst Priority of the VM Instance between 1 and 4.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
metrics_collection	<Boolean>	Read-write	Flag to check if metrics collection is enabled or disabled..
is_managed	<Boolean>	Read-write	Is Managed.
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device. Minimum length = 1 Maximum length = 128
backplane	<String>	Read-write	Backplane Interface. Minimum length = 1
network_interfaces	<network_interface[]>	Read-write	Network Interfaces.
state	<String>	Read-write	Node State. Minimum length = 1 Maximum length = 32
last_updated_time	<Double>	Read-write	Last Updated Time.
license_edition	<String>	Read-write	Edition of instance.
customid	<String>	Read-write	Custom ID.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..
la_mgmt	<Boolean>	Read-write	Bond consisting of management ports on VM Instance.
vlan_id_0_2	<Integer>	Read-write	VLAN id for the management interface 0/2. Maximum value =
number_of_ssl_cores	<Integer>	Read-write	Assign number of ssl virtual functions to VM Instance.
number_of_ssl_cards	<Integer>	Read-write	Number of SSL Cards.
max_burst_throughput	<Double>	Read-write	Maximum burst throughput in Mbps of VM Instance.
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal network (not required to be set for SDWAN).
cmd_policy	<String>	Read-write	true if you want to allow shell/sftp/scp access to Citrix ADC Instance administrator. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 1024
node_id	<String>	Read-write	Node identification of a device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1 Maximum length = 64
throughput_allocation_mode	<Integer>	Read-write	Throughput Allocation Mode: 0-Fixed, 1-Burst-able.
number_of_cores	<Integer>	Read-write	Number of cores that are assigned to VM Instance.
pps	<Double>	Read-write	Assign packets per seconds to Citrix ADC Instance. Maximum value =
plugin_netmask	<String>	Read-write	Signaling Netmask. Minimum length = 1 Maximum length = 64
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be A-B, and for Cluster it will be CLIP. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
std_bw_total	<Integer>	Read-write	Standard Bandwidth.
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1 Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.
vlan_1_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/2 on VM Instance. Maximum value =
vrid_list_ipv4_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV4 VMAC Generation.
vlan_1_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/7 on VM Instance. Maximum value =
vrid_list_ipv6_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 1/2 for IPV6 VMAC Generation.
vrid_list_ipv4_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV4 VMAC Generation.
vrid_list_ipv6_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV6 VMAC Generation.
if_10_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/4 on VM Instance.
vlan_10_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/5 on VM Instance. Maximum value =
vrid_list_ipv6_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV6 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV6 VMAC Generation.
if_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/5 on VM Instance.
receiveuntagged_1_ 4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/4 on VM Instance.
vrid_list_ipv6_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV6 VMAC Generation.
vrid_list_ipv6_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV6 VMAC Generation.
if_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/3 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV6 VMAC Generation.
receiveuntagged_10 _4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/4 on VM Instance.
vrid_list_ipv6_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV6 VMAC Generation.
vlan_10_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/6 on VM Instance. Maximum value =
if_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/5 on VM Instance.
save_config	<Boolean>	Read-write	Should config be saved first in case instance is rebooted while modify.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_new_crypto	<Boolean>	Read-write	True if number_of_acu/ number_of_scu are used, false if number_of_ssl_core s is used.
vrid_list_ipv6_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV6 VMAC Generation.
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
if_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/7 on VM Instance.
receiveuntagged_10 _8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/8 on VM Instance.
vrid_list_ipv6_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 1/4 for IPV6 VMAC Generation.
receiveuntagged_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/6 on VM Instance.
vlan_1_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/4 on VM Instance. Maximum value =
if_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/2 on VM Instance.
receiveuntagged_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/5 on VM Instance.
receiveuntagged_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/6 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/1 on VM Instance.
vrid_list_ipv4_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV4 VMAC Generation.
vrid_list_ipv6_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV6 VMAC Generation.
vrid_list_ipv6_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV6 VMAC Generation.
vrid_list_ipv4_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV4 VMAC Generation.
receiveuntagged_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Untagged Packets on 10/2 on VM Instance.
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
vrid_list_ipv4_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV4 VMAC Generation.
receiveuntagged_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/7 on VM Instance.
vlan_10_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/4 on VM Instance. Maximum value =
if_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/7 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/1 on VM Instance.
vrid_list_ipv6_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV6 VMAC Generation.
if_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/1 on VM Instance.
if_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/8 on VM Instance.
l2_enabled	<Boolean>	Read-write	L2mode status of VM Instance.
vlan_10_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/7 on VM Instance. Maximum value =
vlan_10_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVLAN for Network 10/1 on VM Instance. Maximum value =
receiveuntagged_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/2 on VM Instance.
reboot_vm_on_cpu_change	<Boolean>	Read-write	Reboot VMs on CPU change during resource allocation.
vlan_1_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/6 on VM Instance. Maximum value =
vrid_list_ipv4_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV4 VMAC Generation.
receiveuntagged_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/7 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv4_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV4 VMAC Generation.
vlan_1_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/1 on VM Instance. Maximum value =
vrid_list_ipv6_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV6 VMAC Generation.
vlan_1_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/8 on VM Instance. Maximum value =
vrid_list_ipv4_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/5 on VM Instance.
vlan_10_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/2 on VM Instance. Maximum value =
vrid_list_ipv4_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV4 VMAC Generation.
vrid_list_ipv4_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV4 VMAC Generation.
receiveuntagged_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/8 on VM Instance.
vrid_list_ipv4_10_4	<String[]>	Read-write	This property is deprecated by

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV4 VMAC Generation.
vlan_1_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/3 on VM Instance. Maximum value =
receiveuntagged_10 _3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/3 on VM Instance.
if_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/6 on VM Instance.
if_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/6 on VM Instance.
vlan_10_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/3 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_1_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/5 on VM Instance. Maximum value =
if_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/3 on VM Instance.
nsvlan_interfaces	<String[]>	Read-write	VLAN Interfaces. Minimum length = 1 Maximum length = 50
vrid_list_ipv4_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV4 VMAC Generation.
if_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/4 on VM Instance.
vrid_list_ipv4_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			IPV4 VMAC Generation.
if_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/1 on VM Instance.
vrid_list_ipv4_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV4 VMAC Generation.
vlan_10_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/8 on VM Instance. Maximum value =
receiveuntagged_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/3 on VM Instance.
crypto_change_requires_reboot	<Boolean>	Read-write	True if the current changes made by user requires a reboot of the VM else false.
vrid_list_ipv4_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVRID List for Interface 10/2 for IPV4 VMAC Generation.
if_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/8 on VM Instance.
vrid_list_ipv4_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV4 VMAC Generation.
vrid_list_ipv6_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for IPV6 VMAC Generation.
if_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/2 on VM Instance.
vrid_list_ipv6_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV6 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
clusterid	<Integer>	Read-only	Cluster Id.
instance_state	<String>	Read-only	State of device, UP only if device accessible.
disk_space	<Double>	Read-only	Show Disk Space (GB) available to VM Instance.
diskavail	<Integer>	Read-only	Citrix ADC Disk Available(MB).
cpu_core_pe	<String>	Read-only	Packet Engine cores assigned to VM Instance.
disk1_used	<Double>	Read-only	Disk1 used space in percentage.
clip	<String>	Read-only	Cluster IP Address.
preferred_cpu_socket	<String>	Read-only	Preferred CPU Socket.
device_uuid	<String>	Read-only	Device UUID.
version	<String>	Read-only	Device Version.
vm_memory_used	<Double>	Read-only	Memory Used (MB) by VM Instance.
health	<String>	Read-only	Node Health.
disksize	<Integer>	Read-only	Citrix ADC Disk Size(MB).
sysid	<String>	Read-only	System ID.
nodeid	<Integer>	Read-only	Node Id.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vm_state	<String>	Read-only	State of Virtual Machine (Running | Halted).
uuid	<String>	Read-only	UUID of VM Instance generated by Hypervisor.
disk1_free	<Double>	Read-only	Disk1 free space in MB.
cpu_core_mgmt	<String>	Read-only	Management CPU cores assigned to VM Instance.
ssl_virtual_functions	<String>	Read-only	SSL Virtual Functions assigned to VM Instance.
disk0_total	<Double>	Read-only	Disk0 total space in MB.
http_req	<Double>	Read-only	HTTP Requests/ second.
nic_discards_in	<Double>	Read-only	nic discards on incoming packets.
disk0_free	<Double>	Read-only	Disk0 Free space in MB.
nic_discards_out	<Double>	Read-only	nic discards on outgoing packets.
status	<String>	Read-only	Status of managed device.
systemname	<String>	Read-only	Device System Name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
mac_eth0	<String>	Read-only	MAC Address of eth0 on VM Instance.
ns_memory_usage	<Double>	Read-only	Memory Usage (%).
vm_tx	<Double>	Read-only	Out Throughput of VM Instance in Mbps.
vm_memory_free	<Double>	Read-only	Free Memory (MB) available in VM Instance.
vm_memory_usage	<Double>	Read-only	Memory Usage (%) of VM Instance.
vm_cpu_usage	<Double>	Read-only	CPU Usage (%) of VM Instance.
sdxtools_http_port	<Integer>	Read-only	Http Port number to communicate to SDXTools.
reason	<String>	Read-only	Reason of failure for this managed device.
virtual_functions	<String>	Read-only	Virtual Functions assigned to VM Instance depending on SDX platform.
ns_total_tx	<Double>	Read-only	Total Tx of Citrix ADC Instance in Mbits.
upsince	<String>	Read-only	Upsince of managed device.
domid	<Double>	Read-only	DOM Id assigned to VM Instance viewed in hypervisor.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
tenant_id	<String>	Read-only	Tenant ID.
ns_mgmt_cpu_usage	<Double>	Read-only	Management CPU Usage (%).
do_config	<Boolean>	Read-only	Do default config for managed device.
host_id	<String>	Read-only	Host ID.
has_sdxtools	<Boolean>	Read-only	True if SDX Tools are installed on this VM.
system_hardwareversion	<String>	Read-only	System Hardware Version.
ns_tx	<Double>	Read-only	Out Throughput of Citrix ADC Instance in Mbps.
number_of_cpu	<Integer>	Read-only	Number of CPU that is assigned to VM Instance.
is_swg	<Boolean>	Read-only	Boolean to indicate whether a VM is SWG.
if_0_3	<Boolean>	Read-only	Network 0/3 on VM Instance.
diskused	<Integer>	Read-only	Citrix ADC Disk Used(MB).
uptime	<String>	Read-only	Uptime of device.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.
assigned_cpu_socket	<String>	Read-only	Assigned CPU Socket 0 or 1.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_cpu_usage	<Double>	Read-only	CPU Usage (%) of Citrix ADC Instance.
sdxtools_public_key	<String>	Read-only	SDXTools HTTPS Public Key.
ns_rx	<Double>	Read-only	In Throughput of Citrix ADC Instance in Mbps.
vm_description	<String>	Read-only	Description of the VM instance.
vm_rx	<Double>	Read-only	In Throughput of VM Instance in Mbps.
disk_allocation	<String>	Read-only	Disk allocation for VM Instance.
diskperusage	<Integer>	Read-only	Citrix ADC Disk Utilization (%).
disk1_total	<Double>	Read-only	Disk1 total space in MB.
disk0_used	<Double>	Read-only	Disk0 used space in percentage.
ns_total_rx	<Double>	Read-only	Total Rx of Citrix ADC Instance in Mbits.
sdxtools_version	<String>	Read-only	SDXTools version running on VM Instance.
act_id	<String>	Read-only	Activity Id.
diff_exists	<Boolean>	Read-only	Indicates whether config diff exists for this instances or not.

Operations

(click to see Properties)

- REBOOT
- STOP
- ADD
- PROVISION_CPX
- FORCE_REBOOT
- GET (ALL)
- GET
- DELETE
- UPDATE
- FORCE_STOP
- START

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

REBOOT

URL: https://<MGMT-IP>/nitro/v2/config/ns/id_value<String>?action=reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

STOP

URL: https://<MGMT-IP>/nitro/v2/config/ns/id_value<String>?action=stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns": {

<b>"ip_address":<String_value></b>,
```

```
"vlan_1_2":<Integer_value>,

"ns_ip_address":<String_value>,

"std_bw_config":<Integer_value>,

"password":<String_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"plugin_ip_address":<String_value>,

"vlan_type":<Integer_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"nsvlan_tagged":<Boolean_value>,
```

```
"netmask":<String_value>,

"save_config":<Boolean_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"number_of_ssl_cores_up":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"if_0_1":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,
```

```
"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"is_clip":<Boolean_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,
```

```
"gateway":<String_value>,

"nsvlan_interfaces":<String_value[]>,

"if_1_4":<Boolean_value>,

"iscco":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"license":<String_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"if_0_2":<Boolean_value>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,
```

```
"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"username":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"nsvlan_id":<Integer_value>,

"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"num_pes":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"metrics_collection":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,
```

```
"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"backplane":<String_value>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,
```

```
"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"state":<String_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"vlan_1_6":<Integer_value>,

"license_edition":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,
```

```
"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"customid":<String_value>,

"vlan_1_8":<Integer_value>,

"license_grace_time":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"vrid_list_ipv4_1_8":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"if_10_6":<Boolean_value>,

"receiveuntagged_10_3":<Boolean_value>,

"number_of_ssl_cards":<Integer_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,
```

```
"cmd_policy":<String_value>,

"node_id":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"pps":<Double_value>,

"plugin_netmask":<String_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt, "ns":
[{

"clusterid":<Integer_value>,
```

```
"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"ns_ip_address":<String_value>,

"password":<String_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,

"disk_space":<Double_value>,

"nsvlan_interfaces_db":<String_value>,

"diskavail":<Integer_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"plugin_ip_address":<String_value>,

"vlan_type":<Integer_value>,

"ent_bw_total":<Integer_value>,

"disk1_used":<Double_value>,

"vcpu_config":<Integer_value>,

"clip":<String_value>,

"nsvlan_tagged":<Boolean_value>,

"preferred_cpu_socket":<String_value>,
```

```
"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"health":<String_value>,

"number_of_ssl_cores_up":<Integer_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"disksize":<Integer_value>,

"sysid":<String_value>,

"nodeid":<Integer_value>,

"if_0_1":<Boolean_value>,

"vm_state":<String_value>,

"uuid":<String_value>,

"disk1_free":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,
```

```
"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"disk0_total":<Double_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"is_clip":<Boolean_value>,

"device_family":<String_value>,

"http_req":<Double_value>,

"nic_discards_in":<Double_value>,

"disk0_free":<Double_value>,

"type":<String_value>,

"throughput":<Double_value>,

"template_name":<String_value>,

"nic_discards_out":<Double_value>,

"status":<String_value>,

"gateway":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"ns_memory_usage":<Double_value>,
```

```
"iscco":<Boolean_value>,

"vm_tx":<Double_value>,

"number_of_scu":<Integer_value>,

"license":<String_value>,

"domain_name":<String_value>,

"vm_memory_free":<Double_value>,

"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"if_0_2":<Boolean_value>,

"image_name":<String_value>,

"hostname":<String_value>,

"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ns_total_tx":<Double_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"username":<String_value>,
```

```
"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"user_driven":<Boolean_value>,

"nsvlan_id":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"ns_mgmt_cpu_usage":<Double_value>,

"num_pes":<Integer_value>,

"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"metrics_collection":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"system_hardwareversion":<String_value>,

"ns_tx":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,
```

```
"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"dns":<String_value>,

"backplane":<String_value>,

"if_0_3":<Boolean_value>,

"diskused":<Integer_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,
```

```
"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"state":<String_value>,

"task_id":<String_value>,

"last_updated_time":<Double_value>,

"uptime":<String_value>,

"license_edition":<String_value>,
```

```
"cpu_license_type":<Integer_value>,

"customid":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"ns_cpu_usage":<Double_value>,

"number_of_ssl_cores":<Integer_value>,

"number_of_ssl_cards":<Integer_value>,

"sdxtools_public_key":<String_value>,

"ns_rx":<Double_value>,

"user_profile_name":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"cmd_policy":<String_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"diskperusage":<Integer_value>,
```

```
"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"pps":<Double_value>,

"plugin_netmask":<String_value>,

"disk1_total":<Double_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"disk0_used":<Double_value>,

"nexthop":<String_value>,

"ns_total_rx":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,
```

```
"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"save_config":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,
```

```
"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,
```

```
"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"diff_exists":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vrid_list_ipv4_10_3":<String_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vlan_10_2":<Integer_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,
```

```
"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"nsvlan_interfaces":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,
```

```
"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

PROVISION_CPX

URL: https://<MGMT-IP>/nitro/v2/config/ns?action=provision_cpx;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns": {

"vlan_1_2":<Integer_value>,

"ns_ip_address":<String_value>,

"std_bw_config":<Integer_value>,

"password":<String_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,
```

```
"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"plugin_ip_address":<String_value>,

"mastools_version":<String_value>,

"vlan_type":<Integer_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"nsvlan_tagged":<Boolean_value>,

"netmask":<String_value>,

"save_config":<Boolean_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,
```

```
"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"number_of_ssl_cores_up":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"if_0_1":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"host_ip_address":<String_value>,

"plt_bw_total":<Integer_value>,

"id":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"mgmt_ip_address":<String_value>,
```

```
"ipv6_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"is_clip":<Boolean_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"throughput":<Double_value>,

"type":<String_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"nsvlan_interfaces":<String_value[]>,

"if_1_4":<Boolean_value>,

"iscco":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,
```

```
"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"license":<String_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"if_0_2":<Boolean_value>,

"vrid_list_ipv4_1_2":<String_value[]>,

"vlan_1_7":<Integer_value>,

"image_name":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"hostname":<String_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv6_10_6":<String_value[]>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"username":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"if_10_3":<Boolean_value>,
```

```
"vrid_list_ipv6_1_3":<String_value[]>,

"receiveuntagged_10_4":<Boolean_value>,

"nsvlan_id":<Integer_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"burst_priority":<Integer_value>,

"num_pes":<Integer_value>,

"is_new_crypto":<Boolean_value>,

"metrics_collection":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv6_1_5":<String_value[]>,

"vrid_list_ipv4_1_6":<String_value[]>,

"profile_name":<String_value>,

"backplane":<String_value>,
```

```
"receiveuntagged_1_7":<Boolean_value>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,
```

```
"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"vlan_10_4":<Integer_value>,

"state":<String_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"if_10_8":<Boolean_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"vlan_1_6":<Integer_value>,

"license_edition":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"customid":<String_value>,

"vlan_1_8":<Integer_value>,

"license_grace_time":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,
```

```
"vrid_list_ipv4_1_8":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"if_10_6":<Boolean_value>,

"receiveuntagged_10_3":<Boolean_value>,

"number_of_ssl_cards":<Integer_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"cmd_policy":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,
```

```
"plugin_netmask":<String_value>,

"pps":<Double_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt, "ns":
[{

"clusterid":<Integer_value>,

"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"ns_ip_address":<String_value>,

"password":<String_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,
```

```
"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,

"disk_space":<Double_value>,

"nsvlan_interfaces_db":<String_value>,

"diskavail":<Integer_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"plugin_ip_address":<String_value>,

"vlan_type":<Integer_value>,

"ent_bw_total":<Integer_value>,

"disk1_used":<Double_value>,

"vcpu_config":<Integer_value>,

"clip":<String_value>,

"nsvlan_tagged":<Boolean_value>,

"preferred_cpu_socket":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,
```

```
"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"health":<String_value>,

"number_of_ssl_cores_up":<Integer_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"disksize":<Integer_value>,

"sysid":<String_value>,

"nodeid":<Integer_value>,

"if_0_1":<Boolean_value>,

"vm_state":<String_value>,

"uuid":<String_value>,

"disk1_free":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"disk0_total":<Double_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,
```

```
"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"is_clip":<Boolean_value>,

"device_family":<String_value>,

"http_req":<Double_value>,

"nic_discards_in":<Double_value>,

"disk0_free":<Double_value>,

"type":<String_value>,

"throughput":<Double_value>,

"template_name":<String_value>,

"nic_discards_out":<Double_value>,

"status":<String_value>,

"gateway":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"ns_memory_usage":<Double_value>,

"iscco":<Boolean_value>,

"vm_tx":<Double_value>,

"number_of_scu":<Integer_value>,

"license":<String_value>,

"domain_name":<String_value>,

"vm_memory_free":<Double_value>,
```

```
"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"if_0_2":<Boolean_value>,

"image_name":<String_value>,

"hostname":<String_value>,

"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ns_total_tx":<Double_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"username":<String_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"user_driven":<Boolean_value>,

"nsvlan_id":<Integer_value>,
```

```
"host_type":<String_value>,

"tenant_id":<String_value>,

"ns_mgmt_cpu_usage":<Double_value>,

"num_pes":<Integer_value>,

"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"metrics_collection":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"system_hardwareversion":<String_value>,

"ns_tx":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"dns":<String_value>,

"backplane":<String_value>,

"if_0_3":<Boolean_value>,

"diskused":<Integer_value>,
```

```
"network_interfaces":[{
"device_channel_name":<String_value>,
"is_mgmt_ifc":<Boolean_value>,
"name_server":<String_value>,
"vrid_list_ipv6":<String_value>,
"gateway":<String_value>,
"channel_tag_all_vlans":<Boolean_value>,
"xen_vf_index":<Integer_value>,
"parent_id":<String_value>,
"vrid_list_ipv4":<String_value>,
"is_member_ifc":<Boolean_value>,
"mac_address":<String_value>,
"parent_channel_id":<String_value>,
"channel_type":<String_value>,
"l2_enabled":<Boolean_value>,
"id":<String_value>,
"ip_address":<String_value>,
"netmask":<String_value>,
"port_name":<String_value>,
"parent_name":<String_value>,
"channel_timeout":<String_value>,
"mac_mode":<String_value>,
```

```
"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"state":<String_value>,

"task_id":<String_value>,

"last_updated_time":<Double_value>,

"uptime":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"customid":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,
```

```
"ns_cpu_usage":<Double_value>,

"number_of_ssl_cores":<Integer_value>,

"number_of_ssl_cards":<Integer_value>,

"sdxtools_public_key":<String_value>,

"ns_rx":<Double_value>,

"user_profile_name":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"cmd_policy":<String_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"diskperusage":<Integer_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"pps":<Double_value>,

"plugin_netmask":<String_value>,
```

```
"disk1_total":<Double_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"disk0_used":<Double_value>,

"nexthop":<String_value>,

"ns_total_rx":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,
```

```
"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"save_config":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,
```

```
"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"diff_exists":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,
```

```
"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vrid_list_ipv4_10_3":<String_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vlan_10_2":<Integer_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,
```

```
"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"nsvlan_interfaces":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

```

```

FORCE_REBOOT

URL: https://<MGMT-IP>/nitro/v2/config/ns/id_value<String>?action=force_reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns **?count=yes** Use this query-parameter to get the count of ns resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt, "ns":
[{

"clusterid":<Integer_value>,

"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"ns_ip_address":<String_value>,

"password":<String_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,
```

```
"disk_space":<Double_value>,

"nsvlan_interfaces_db":<String_value>,

"diskavail":<Integer_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"plugin_ip_address":<String_value>,

"vlan_type":<Integer_value>,

"ent_bw_total":<Integer_value>,

"disk1_used":<Double_value>,

"vcpu_config":<Integer_value>,

"clip":<String_value>,

"nsvlan_tagged":<Boolean_value>,

"preferred_cpu_socket":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"health":<String_value>,
```

```
"number_of_ssl_cores_up":<Integer_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"disksize":<Integer_value>,

"sysid":<String_value>,

"nodeid":<Integer_value>,

"if_0_1":<Boolean_value>,

"vm_state":<String_value>,

"uuid":<String_value>,

"disk1_free":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"disk0_total":<Double_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"is_clip":<Boolean_value>,
```

```
"device_family":<String_value>,

"http_req":<Double_value>,

"nic_discards_in":<Double_value>,

"disk0_free":<Double_value>,

"type":<String_value>,

"throughput":<Double_value>,

"template_name":<String_value>,

"nic_discards_out":<Double_value>,

"status":<String_value>,

"gateway":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"ns_memory_usage":<Double_value>,

"iscco":<Boolean_value>,

"vm_tx":<Double_value>,

"number_of_scu":<Integer_value>,

"license":<String_value>,

"domain_name":<String_value>,

"vm_memory_free":<Double_value>,

"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"if_0_2":<Boolean_value>,
```

```
"image_name":<String_value>,

"hostname":<String_value>,

"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ns_total_tx":<Double_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"username":<String_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"user_driven":<Boolean_value>,

"nsvlan_id":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"ns_mgmt_cpu_usage":<Double_value>,
```

```
"num_pes":<Integer_value>,

"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"metrics_collection":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"system_hardwareversion":<String_value>,

"ns_tx":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"dns":<String_value>,

"backplane":<String_value>,

"if_0_3":<Boolean_value>,

"diskused":<Integer_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,
```

```
"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,
```

```
"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"state":<String_value>,

"task_id":<String_value>,

"last_updated_time":<Double_value>,

"uptime":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"customid":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"ns_cpu_usage":<Double_value>,

"number_of_ssl_cores":<Integer_value>,

"number_of_ssl_cards":<Integer_value>,
```

```
"sdxtools_public_key":<String_value>,

"ns_rx":<Double_value>,

"user_profile_name":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"cmd_policy":<String_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"diskperusage":<Integer_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"pps":<Double_value>,

"plugin_netmask":<String_value>,

"disk1_total":<Double_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,
```

```
"disk0_used":<Double_value>,

"nexthop":<String_value>,

"ns_total_rx":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,
```

```
"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"save_config":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,
```

```
"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"diff_exists":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,
```

```
"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vrid_list_ipv4_10_3":<String_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vlan_10_2":<Integer_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,
```

```
"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"nsvlan_interfaces":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```


GET

URL: https://<MGMT-IP>/nitro/v2/config/ns/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt, "ns":
[{

"clusterid":<Integer_value>,

"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"ns_ip_address":<String_value>,

"password":<String_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,

"disk_space":<Double_value>,

"nsvlan_interfaces_db":<String_value>,
```

```
"diskavail":<Integer_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"plugin_ip_address":<String_value>,

"vlan_type":<Integer_value>,

"ent_bw_total":<Integer_value>,

"disk1_used":<Double_value>,

"vcpu_config":<Integer_value>,

"clip":<String_value>,

"nsvlan_tagged":<Boolean_value>,

"preferred_cpu_socket":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"health":<String_value>,

"number_of_ssl_cores_up":<Integer_value>,

"std_bw_available":<Integer_value>,
```

```
"internal_ip_address":<String_value>,

"disksize":<Integer_value>,

"sysid":<String_value>,

"nodeid":<Integer_value>,

"if_0_1":<Boolean_value>,

"vm_state":<String_value>,

"uuid":<String_value>,

"disk1_free":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"disk0_total":<Double_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"is_clip":<Boolean_value>,

"device_family":<String_value>,

"http_req":<Double_value>,
```

```
"nic_discards_in":<Double_value>,

"disk0_free":<Double_value>,

"type":<String_value>,

"throughput":<Double_value>,

"template_name":<String_value>,

"nic_discards_out":<Double_value>,

"status":<String_value>,

"gateway":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"ns_memory_usage":<Double_value>,

"iscco":<Boolean_value>,

"vm_tx":<Double_value>,

"number_of_scu":<Integer_value>,

"license":<String_value>,

"domain_name":<String_value>,

"vm_memory_free":<Double_value>,

"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"if_0_2":<Boolean_value>,

"image_name":<String_value>,

"hostname":<String_value>,
```

```
"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ns_total_tx":<Double_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"username":<String_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"user_driven":<Boolean_value>,

"nsvlan_id":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"ns_mgmt_cpu_usage":<Double_value>,

"num_pes":<Integer_value>,

"burst_priority":<Integer_value>,
```

```
"do_config":<Boolean_value>,

"metrics_collection":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"system_hardwareversion":<String_value>,

"ns_tx":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"dns":<String_value>,

"backplane":<String_value>,

"if_0_3":<Boolean_value>,

"diskused":<Integer_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,
```

```
"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,
```

```
"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"state":<String_value>,

"task_id":<String_value>,

"last_updated_time":<Double_value>,

"uptime":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"customid":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"ns_cpu_usage":<Double_value>,

"number_of_ssl_cores":<Integer_value>,

"number_of_ssl_cards":<Integer_value>,

"sdxtools_public_key":<String_value>,

"ns_rx":<Double_value>,
```

```
"user_profile_name":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"cmd_policy":<String_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"diskperusage":<Integer_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"pps":<Double_value>,

"plugin_netmask":<String_value>,

"disk1_total":<Double_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"disk0_used":<Double_value>,

"nexthop":<String_value>,
```

```
"ns_total_rx":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,
```

```
"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"save_config":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,
```

```
"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"diff_exists":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,
```

```
"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vrid_list_ipv4_10_3":<String_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vlan_10_2":<Integer_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,
```

```
"nsvlan_interfaces":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/ns/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/ns/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns":{

<b>"id":<String_value></b>,

"vlan_1_2":<Integer_value>,

"ns_ip_address":<String_value>,

"std_bw_config":<Integer_value>,

"password":<String_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,
```

```
"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"plugin_ip_address":<String_value>,

"vlan_type":<Integer_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"nsvlan_tagged":<Boolean_value>,

"netmask":<String_value>,

"save_config":<Boolean_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,
```

```
"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"number_of_ssl_cores_up":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"if_0_1":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,
```

```
"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"is_clip":<Boolean_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"nsvlan_interfaces":<String_value[]>,

"if_1_4":<Boolean_value>,

"iscco":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,
```

```
"license":<String_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"if_0_2":<Boolean_value>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"username":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"nsvlan_id":<Integer_value>,
```

```
"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"num_pes":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"metrics_collection":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"backplane":<String_value>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,
```

```
"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,
```

```
"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"state":<String_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"vlan_1_6":<Integer_value>,

"license_edition":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"customid":<String_value>,

"vlan_1_8":<Integer_value>,

"license_grace_time":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"vrid_list_ipv4_1_8":<String_value[]>,

"la_mgmt":<Boolean_value>,
```

```
"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"if_10_6":<Boolean_value>,

"receiveuntagged_10_3":<Boolean_value>,

"number_of_ssl_cards":<Integer_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"cmd_policy":<String_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"pps":<Double_value>,

"plugin_netmask":<String_value>,
```

```
"vrid_list_ipv4_10_2":<String_value[]>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt, "ns":
[{

"clusterid":<Integer_value>,

"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"ns_ip_address":<String_value>,

"password":<String_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,
```

```
"name":<String_value>,

"disk_space":<Double_value>,

"nsvlan_interfaces_db":<String_value>,

"diskavail":<Integer_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"plugin_ip_address":<String_value>,

"vlan_type":<Integer_value>,

"ent_bw_total":<Integer_value>,

"disk1_used":<Double_value>,

"vcpu_config":<Integer_value>,

"clip":<String_value>,

"nsvlan_tagged":<Boolean_value>,

"preferred_cpu_socket":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,
```

```
"health":<String_value>,

"number_of_ssl_cores_up":<Integer_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"disksize":<Integer_value>,

"sysid":<String_value>,

"nodeid":<Integer_value>,

"if_0_1":<Boolean_value>,

"vm_state":<String_value>,

"uuid":<String_value>,

"disk1_free":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"disk0_total":<Double_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,
```

```
"is_clip":<Boolean_value>,

"device_family":<String_value>,

"http_req":<Double_value>,

"nic_discards_in":<Double_value>,

"disk0_free":<Double_value>,

"type":<String_value>,

"throughput":<Double_value>,

"template_name":<String_value>,

"nic_discards_out":<Double_value>,

"status":<String_value>,

"gateway":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"ns_memory_usage":<Double_value>,

"iscco":<Boolean_value>,

"vm_tx":<Double_value>,

"number_of_scu":<Integer_value>,

"license":<String_value>,

"domain_name":<String_value>,

"vm_memory_free":<Double_value>,

"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,
```

```
"if_0_2":<Boolean_value>,

"image_name":<String_value>,

"hostname":<String_value>,

"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ns_total_tx":<Double_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"username":<String_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"user_driven":<Boolean_value>,

"nsvlan_id":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,
```

```
"ns_mgmt_cpu_usage":<Double_value>,

"num_pes":<Integer_value>,

"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"metrics_collection":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"system_hardwareversion":<String_value>,

"ns_tx":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"dns":<String_value>,

"backplane":<String_value>,

"if_0_3":<Boolean_value>,

"diskused":<Integer_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,
```

```
"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,
```

```
"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"state":<String_value>,

"task_id":<String_value>,

"last_updated_time":<Double_value>,

"uptime":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"customid":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"ns_cpu_usage":<Double_value>,

"number_of_ssl_cores":<Integer_value>,
```

```
"number_of_ssl_cards":<Integer_value>,

"sdxtools_public_key":<String_value>,

"ns_rx":<Double_value>,

"user_profile_name":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"cmd_policy":<String_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"diskperusage":<Integer_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"pps":<Double_value>,

"plugin_netmask":<String_value>,

"disk1_total":<Double_value>,

"display_name":<String_value>,
```

```
"std_bw_total":<Integer_value>,

"disk0_used":<Double_value>,

"nexthop":<String_value>,

"ns_total_rx":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,
```

```
"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"save_config":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,
```

```
"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"diff_exists":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,
```

```
"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vrid_list_ipv4_10_3":<String_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vlan_10_2":<Integer_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,
```

```
"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"nsvlan_interfaces":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

FORCE_STOP

URL: https://<MGMT-IP>/nitro/v2/config/ns/id_value<String>?action=force_stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

START

URL: https://<MGMT-IP>/nitro/v2/config/ns/id_value<String>?action=start;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

ns_appflow_param_config

Configuration for Appflow param configuration on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
observationDomainName	<String>	Read-write	Citrix ADC observationDomainName. Minimum length = 1 Maximum length = 256
is_clip	<Boolean>	Read-write	Is Clip.
template_interval	<Integer>	Read-write	Template refresh interval.
httpxforwardedfor	<String>	Read-write	. Minimum length = 1 Maximum length = 10
observationDomainID	<Double>	Read-write	observationDomainID.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- GET (ALL)
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_appflow_param_config **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_appflow_param_config **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_appflow_param_config resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_appflow_param_config **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_appflow_param_config resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_appflow_param_config **?count=yes** Use this query-parameter to get the count of ns_appflow_param_config resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_appflow_param_config":[{

"observationDomainName":<String_value>,

"is_clip":<Boolean_value>,

"template_interval":<Integer_value>,

"httpxforwardedfor":<String_value>,

"observationDomainID":<Double_value>,

"ns_ip_address":<String_value>}]}
```

MODIFY

URL: https://<MGMT-IP>/nitro/v2/config/ns_appflow_param_config/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_appflow_param_config":{

"ns_ip_address":<String_value>,

"is_clip":<Boolean_value>,

"template_interval":<Integer_value>,

"observationDomainID":<Double_value>,

"observationDomainName":<String_value>,
```

```
"httpxforwardedfor":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_appflow_param_config":[{

"observationDomainName":<String_value>,

"is_clip":<Boolean_value>,

"template_interval":<Integer_value>,

"httpxforwardedfor":<String_value>,

"observationDomainID":<Double_value>,

"ns_ip_address":<String_value>}]}
```

ns_callhome

Configuration for Citrix ADC Callhome Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
proxy_mode	<String>	Read-write	Deploy the callhome proxy mode.Possible values YES or NO. Maximum length = 78

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
proxy_server_ip	<String>	Read-write	Proxy Server IP address. Minimum length = 1 Maximum length = 64
ip_address	<String>	Read-write	Citrix ADC IP Address for callhome. Minimum length = 1 Maximum length = 64
proxy_server_port	<Integer>	Read-write	Proxy server Port. Maximum value =
email_address	<String>	Read-write	The contact person's E-mail address. Maximum length = 78
name	<String>	Read-write	Name of the vpx Instance. Maximum length = 78
ns_ip_address_arr	<String[]>	Read-write	List of Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
powfirstfail	<String>	Read-only	First occurrence power supply unit failure.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
callhomesenablestatus	<String>	Read-only	Callhome feature enabled/disable.
powlatestfail	<String>	Read-only	Latest occurrence power supply unit failure.
flashlatestfail	<String>	Read-only	Latest occurrence compact flash failure.
restartfirstfail	<String>	Read-only	First occurrence warm restart failure.
hddlatestfail	<String>	Read-only	Latest occurrence hard disk drive failure.
flashfirstfail	<String>	Read-only	First occurrence compact flash failure.
restartlatestfail	<String>	Read-only	Latest occurrence warm restart failure.
hddfirstfail	<String>	Read-only	First occurrence hard disk drive failure.
callhomestatus	<String>	Read-only	Callhome feature register with upload server successful/ failed.
sslcardfirstfailure	<String>	Read-only	First occurrence SSL card failure.
sslcardlatestfailure	<String>	Read-only	Latest occurrence SSL card failure.

Operations

(click to see Properties)

- POLLNOW

- ENABLE
- GET (ALL)
- UPDATE
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

POLLNOW

URL: https://<MGMT-IP>/nitro/v2/config/ns_callhome?action=pollnow;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_callhome": {

<b>"ip_address":<String_value></b>,

"proxy_mode":<String_value>,

"name":<String_value>,

"email_address":<String_value>,

"ns_ip_address_arr":<String_value[]>,

"proxy_server_ip":<String_value>,

"proxy_server_port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_callhome":[{

"powfirstfail":<String_value>,
```

```
"callhomesenablestatus":<String_value>,

"proxy_mode":<String_value>,

"powlatestfail":<String_value>,

"flashlatestfail":<String_value>,

"restartfirstfail":<String_value>,

"hddlatestfail":<String_value>,

"proxy_server_ip":<String_value>,

"flashfirstfail":<String_value>,

"ip_address":<String_value>,

"restartlatestfail":<String_value>,

"proxy_server_port":<Integer_value>,

"hddfirstfail":<String_value>,

"email_address":<String_value>,

"callhomestatus":<String_value>,

"name":<String_value>,

"ns_ip_address_arr":<String_value>,

"sslcardfirstfailure":<String_value>,

"sslcardlatestfailure":<String_value>}]}
```

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_callhome?action=enable;onerror=<String_value>

HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_callhome": {

<b>"ip_address":<String_value></b>,

"proxy_mode":<String_value>,

"name":<String_value>,

"email_address":<String_value>,

"ns_ip_address_arr":<String_value[]>,

"proxy_server_ip":<String_value>,

"proxy_server_port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_callhome":[{

"powfirstfail":<String_value>,

"callhomesenablestatus":<String_value>,

"proxy_mode":<String_value>,

"powlatestfail":<String_value>,
```

```
"flashlatestfail":<String_value>,

"restartfirstfail":<String_value>,

"hddlatestfail":<String_value>,

"proxy_server_ip":<String_value>,

"flashfirstfail":<String_value>,

"ip_address":<String_value>,

"restartlatestfail":<String_value>,

"proxy_server_port":<Integer_value>,

"hddfirstfail":<String_value>,

"email_address":<String_value>,

"callhomestatus":<String_value>,

"name":<String_value>,

"ns_ip_address_arr":<String_value>,

"sslcardfirstfailure":<String_value>,

"sslcardlatestfailure":<String_value>}]}
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_callhome **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_callhome **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_callhome resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_callhome **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_callhome resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_callhome **?count=yes** Use this query-parameter to get the count of ns_callhome resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_callhome":[{

"powfirstfail":<String_value>,

"callhomesenablestatus":<String_value>,

"proxy_mode":<String_value>,

"powlatestfail":<String_value>,

"flashlatestfail":<String_value>,

"restartfirstfail":<String_value>,
```

```
"hddlatestfail":<String_value>,

"proxy_server_ip":<String_value>,

"flashfirstfail":<String_value>,

"ip_address":<String_value>,

"restartlatestfail":<String_value>,

"proxy_server_port":<Integer_value>,

"hddfirstfail":<String_value>,

"email_address":<String_value>,

"callhomestatus":<String_value>,

"name":<String_value>,

"ns_ip_address_arr":<String_value>,

"sslcardfirstfailure":<String_value>,

"sslcardlatestfailure":<String_value>}]}
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/ns_callhome/ip_address_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_callhome":{

"proxy_mode":<String_value>,

"name":<String_value>,

"email_address":<String_value>,

"ns_ip_address_arr":<String_value[]>,

"proxy_server_ip":<String_value>,

"proxy_server_port":<Integer_value>,

"ip_address":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_callhome":[{

"powfirstfail":<String_value>,

"callhomesenablestatus":<String_value>,

"proxy_mode":<String_value>,

"powlatestfail":<String_value>,

"flashlatestfail":<String_value>,

"restartfirstfail":<String_value>,

"hddlatestfail":<String_value>,

"proxy_server_ip":<String_value>,
```

```
"flashfirstfail":<String_value>,

"ip_address":<String_value>,

"restartlatestfail":<String_value>,

"proxy_server_port":<Integer_value>,

"hddfirstfail":<String_value>,

"email_address":<String_value>,

"callhomestatus":<String_value>,

"name":<String_value>,

"ns_ip_address_arr":<String_value>,

"sslcardfirstfailure":<String_value>,

"sslcardlatestfailure":<String_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_callhome?action=disable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_callhome": {

<b>"ip_address":<String_value></b>,
```

```
"proxy_mode":<String_value>,

"name":<String_value>,

"email_address":<String_value>,

"ns_ip_address_arr":<String_value[]>,

"proxy_server_ip":<String_value>,

"proxy_server_port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_callhome":[{

"powfirstfail":<String_value>,

"callhomesenablestatus":<String_value>,

"proxy_mode":<String_value>,

"powlatestfail":<String_value>,

"flashlatestfail":<String_value>,

"restartfirstfail":<String_value>,

"hddlatestfail":<String_value>,

"proxy_server_ip":<String_value>,

"flashfirstfail":<String_value>,

"ip_address":<String_value>,
```

```
"restartlatestfail":<String_value>,

"proxy_server_port":<Integer_value>,

"hddfirstfail":<String_value>,

"email_address":<String_value>,

"callhomestatus":<String_value>,

"name":<String_value>,

"ns_ip_address_arr":<String_value>,

"sslcardfirstfailure":<String_value>,

"sslcardlatestfailure":<String_value>}]}
```

ns_clag_interface

Configuration for CLAG Interface resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
link_redundancy	<Boolean>	Read-write	Link Redundancy. Possible values: ON(for true) and OFF(for false).
channel_tag_all_vlans	<Boolean>	Read-write	If true then all the member interfaces of this channel are tagged. Possible values: true and false.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
lacp_channel_time	<String>	Read-write	LACP time. Possible values: long and short. Minimum length = 4 Maximum length = 10
channel_interface_list	<ns_clag_node[]>	Read-write	Array of CLAG nodes that are part of this channel (10/1, 10/4).
clip	<String>	Read-write	Cluster IPAddress. Minimum length = 1 Maximum length = 64
channel_alias	<String>	Read-write	Alias name for this channel. Minimum length = 4 Maximum length = 31
id	<String>	Read-write	Id is system generated key for CLAG.
mtu	<Integer>	Read-write	MTU value, should be between 1500-9216. Minimum value = 1500 Maximum value =
channel_id	<String>	Read-write	Channel ID if this interface represents

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			a channel (CLA/1, CLA/2 ...). Minimum length = 5 Maximum length = 5
channel_ha_monitoring	<Boolean>	Read-write	HA-monitoring control for the channel. Possible values: true and false.
lr_min_throughput	<Double>	Read-write	LR Minimum Throughout in Mbps. Maximum value =
has_foreign_interfaces	<Boolean>	Read-write	Indicates whether CLAG has interfaces from remote SDX. Possible values: true and false.
sync_operation	<Boolean>	Read-write	sync operation.
lacp_channel_mac_address	<String>	Read-only	Mac address of the bond on cluster.
channel_type	<String>	Read-only	Channel type to represent the CLAG type.
channel_state	<String>	Read-only	Indicates CLA state. Possible values: UP and DOWN.
lacp_mode	<String>	Read-only	LACP mode can be either ACTIVE, PASSIVE, DISABLED.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Applicable for channel.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns_clag_interface?onerror=<String_value> **HTTPS**
Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_clag_interface": {

<b>"channel_interface_list":[{

"nodeid":<Integer_value>,

"parent_id":<String_value>,

"id":<String_value>,

"parent_name":<String_value>,

"interface_name":<String_value>,

"local_node":<Boolean_value>}]</b>,

<b>"clip":<String_value></b>,

<b>"channel_id":<String_value></b>,
```

```
"lr_min_throughput":<Double_value>,

"has_foreign_interfaces":<Boolean_value>,

"lacp_channel_time":<String_value>,

"id":<String_value>,

"sync_operation":<Boolean_value>,

"channel_tag_all_vlans":<Boolean_value>,

"mtu":<Integer_value>,

"link_redundancy":<Boolean_value>,

"channel_alias":<String_value>,

"channel_ha_monitoring":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_clag_interface":[{

"link_redundancy":<Boolean_value>,

"channel_tag_all_vlans":<Boolean_value>,

"lacp_channel_time":<String_value>,

"host_type":<String_value>,

"channel_interface_list":[{

"local_node":<Boolean_value>,
```

```
"interface_name":<String_value>,

"parent_name":<String_value>,

"nodeid":<Integer_value>,

"id":<String_value>,

"parent_id":<String_value>}],

"clip":<String_value>,

"channel_alias":<String_value>,

"host_ip_address":<String_value>,

"lacp_channel_mac_address":<String_value>,

"id":<String_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_id":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"channel_state":<String_value>,

"lacp_mode":<String_value>,

"lr_min_throughput":<Double_value>,

"has_foreign_interfaces":<Boolean_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/ns_clag_interface/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_clag_interface **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_clag_interface **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_clag_interface resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_clag_interface **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_clag_interface resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_clag_interface **?count=yes** Use this query-parameter to get the count of ns_clag_interface resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_clag_interface":[{

"link_redundancy":<Boolean_value>,

"channel_tag_all_vlans":<Boolean_value>,

"lacp_channel_time":<String_value>,

"host_type":<String_value>,

"channel_interface_list":[{

"local_node":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"nodeid":<Integer_value>,

"id":<String_value>,

"parent_id":<String_value>}],

"clip":<String_value>,

"channel_alias":<String_value>,

"host_ip_address":<String_value>,
```

```
"lacp_channel_mac_address":<String_value>,

"id":<String_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_id":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"channel_state":<String_value>,

"lacp_mode":<String_value>,

"lr_min_throughput":<Double_value>,

"has_foreign_interfaces":<Boolean_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_clag_interface/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_clag_interface":[{

"link_redundancy":<Boolean_value>,

"channel_tag_all_vlans":<Boolean_value>,

"lacp_channel_time":<String_value>,

"host_type":<String_value>,

"channel_interface_list":[{

"local_node":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"nodeid":<Integer_value>,

"id":<String_value>,

"parent_id":<String_value>}],

"clip":<String_value>,

"channel_alias":<String_value>,

"host_ip_address":<String_value>,

"lacp_channel_mac_address":<String_value>,

"id":<String_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,
```

```
"channel_id":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"channel_state":<String_value>,

"lacp_mode":<String_value>,

"lr_min_throughput":<Double_value>,

"has_foreign_interfaces":<Boolean_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>}]}
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/ns_clag_interface/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_clag_interface":{

<b>"id":<String_value></b>,

"clip":<String_value>,

"lr_min_throughput":<Double_value>,

"has_foreign_interfaces":<Boolean_value>,

"lacp_channel_time":<String_value>,
```

```
"sync_operation":<Boolean_value>,

"channel_interface_list":[{

"nodeid":<Integer_value>,

"parent_id":<String_value>,

"id":<String_value>,

"parent_name":<String_value>,

"interface_name":<String_value>,

"local_node":<Boolean_value>}],

"channel_tag_all_vlans":<Boolean_value>,

"mtu":<Integer_value>,

"link_redundancy":<Boolean_value>,

"channel_alias":<String_value>,

"channel_id":<String_value>,

"channel_ha_monitoring":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_clag_interface":[{

"link_redundancy":<Boolean_value>,

"channel_tag_all_vlans":<Boolean_value>,
```

```
"lacp_channel_time":<String_value>,

"host_type":<String_value>,

"channel_interface_list":[{

"local_node":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"nodeid":<Integer_value>,

"id":<String_value>,

"parent_id":<String_value>}],

"clip":<String_value>,

"channel_alias":<String_value>,

"host_ip_address":<String_value>,

"lacp_channel_mac_address":<String_value>,

"id":<String_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_id":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"channel_state":<String_value>,

"lacp_mode":<String_value>,

"lr_min_throughput":<Double_value>,

"has_foreign_interfaces":<Boolean_value>,
```

```
"act_id":<String_value>,

"sync_operation":<Boolean_value>}]}
```

ns_clag_node

Configuration for CLAG node resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
local_node	<Boolean>	Read-write	Flag which represents either a local node or foreign node.
interface_name	<String>	Read-write	Name of the interface.
parent_name	<String>	Read-write	.
nodeid	<Integer>	Read-write	Node Id applicable on cluster. Maximum value =
id	<String>	Read-write	Id is system generated key..
parent_id	<String>	Read-write	.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_clag_node **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_clag_node **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_clag_node resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_clag_node **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_clag_node resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_clag_node **?count=yes** Use this query-parameter to get the count of ns_clag_node resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_clag_node":[{

"local_node":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"nodeid":<Integer_value>,

"id":<String_value>,

"parent_id":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_clag_node/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_clag_node":[{

"local_node":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"nodeid":<Integer_value>,

"id":<String_value>,

"parent_id":<String_value>}]}
```

ns_cluster

Configuration for Cluster information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
add_nodegroup	<Boolean>	Read-write	Add Node group in cluster instance if this property is true.
backplane	<String>	Read-write	Backplane Interface.
clusterid	<Integer>	Read-write	Cluster Id in range of 1 and 16. Minimum value = 1 Maximum value =
nodeid	<Integer>	Read-write	Node Id in range of 0 and 31. Maximum value =
state	<String>	Read-write	Node State. Can be ACTIVE, SPARE, PASSIVE. Minimum length = 1 Maximum length = 32
password	<String>	Read-write	Password of the Configuration Coordinator node(Cluster IP). Minimum length = 1 Maximum length = 128
process_local	<Boolean>	Read-write	Process Local.
inc_mode	<Boolean>	Read-write	Inc Mode.
snip	<String>	Read-write	Snip.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
scheduleId	<String>	Read-write	scheduleId is used to refer maintenaince schedule task.
clnodes	<String>	Read-write	List of Nodes part of Cluster.
preemption	<Boolean>	Read-write	Preemption.
iscco	<Boolean>	Read-write	Is Cluster coordinator.
clip	<String>	Read-write	Cluster IPAddress. Minimum length = 1 Maximum length = 64
node_group	<String>	Read-write	Node group.
nodegroup_list	<String[]>	Read-write	Available Node group list on this cluster.
ipaddress	<String>	Read-write	Citrix ADC ipaddress which is going to participate in cluster. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Profile Name. Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
mgmt_cpu_usage	<Double>	Read-only	Management CPU Usage (%) of Cluster Instance.
channel_interface_list	<ns_clag_node[]>	Read-only	Array of CLAG nodes that are part of this channel (10/1, 10/4).
totpropagationtimeout	<Double>	Read-only	Number of times the update to the client timed-out.
totsteeredpkts	<Double>	Read-only	Total number of packets steered on the cluster backplane.
clag_intf_list	<String[]>	Read-only	CLAG Interface List containing interfaces which aren't shared across nodes and aren't backplanes.
act_id	<String>	Read-only	Activity Id.
numdfddroppkts	<Double>	Read-only	Total number of packets steered on the cluster backplane.
clviewleader	<String>	Read-only	Cluster CCO NSIP.
clnumnodes	<Integer>	Read-only	Number of nodes in cluster.
http_req	<Double>	Read-only	HTTP Requests/ second on Cluster Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cpu_usage	<Double>	Read-only	CPU Usage (%) of Cluster Instance.
clbkplanerx	<Double>	Read-only	Traffic received on backplane (in mbits).
clbkplanetx	<Double>	Read-only	Traffic transmitted from backplane (in mbits).
operationalstate	<String>	Read-only	Cluster Operational State.
tx	<Double>	Read-only	Out Throughput of Cluster Instance in Mbps.
clcurstatus	<String>	Read-only	Current State of the cluster.
rx	<Double>	Read-only	In Throughput of Cluster Instance in Mbps.
memory_usage	<Double>	Read-only	Memory Usage (%) of Cluster Instance.
clbkplanetxrate	<Double>	Read-only	Traffic rate transmitted from backplane (in mbits).
health	<String>	Read-only	Node Health State.
clbkplanerxrate	<Double>	Read-only	Traffic rate received on backplane (in mbits).

Operations

(click to see Properties)

- REMOVE

- GET (ALL)
- MODIFY_PASSWORD
- FORM_CLUSTER
- JOIN

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

REMOVE

URL: https://<MGMT-IP>/nitro/v2/config/ns_cluster?action=remove;onerror=<String_value>

HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_cluster": {

<b>"ipaddress":<String_value></b>,

"clusterid":<Integer_value>,

"password":<String_value>,

"process_local":<Boolean_value>,

"clip":<String_value>,

"node_group":<String_value>,

"nodeid":<Integer_value>,

"snip":<String_value>,

"preemption":<Boolean_value>,

"iscco":<Boolean_value>,

"nodegroup_list":<String_value[]>,

"clnodes":<String_value>,

"profile_name":<String_value>,
```

```
"add_nodegroup":<Boolean_value>,

"backplane":<String_value>,

"state":<String_value>,

"inc_mode":<Boolean_value>,

"scheduleId":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_cluster":[{

"add_nodegroup":<Boolean_value>,

"mgmt_cpu_usage":<Double_value>,

"backplane":<String_value>,

"clusterid":<Integer_value>,

"nodeid":<Integer_value>,

"state":<String_value>,

"channel_interface_list":[{

"local_node":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"nodeid":<Integer_value>,
```

```
"id":<String_value>,

"parent_id":<String_value>}],

"totpropagationtimeout":<Double_value>,

"totsteeredpkts":<Double_value>,

"password":<String_value>,

"clag_intf_list":<String_value>,

"process_local":<Boolean_value>,

"act_id":<String_value>,

"numdfddroppkts":<Double_value>,

"clviewleader":<String_value>,

"inc_mode":<Boolean_value>,

"snip":<String_value>,

"clnumnodes":<Integer_value>,

"http_req":<Double_value>,

"scheduleId":<String_value>,

"sync_operation":<Boolean_value>,

"cpu_usage":<Double_value>,

"clbkplanerx":<Double_value>,

"clnodes":<String_value>,

"clbkplanetx":<Double_value>,

"preemption":<Boolean_value>,

"iscco":<Boolean_value>,
```

```
"clip":<String_value>,

"operationalstate":<String_value>,

"tx":<Double_value>,

"clcurstatus":<String_value>,

"rx":<Double_value>,

"memory_usage":<Double_value>,

"node_group":<String_value>,

"clbkplanetxrate":<Double_value>,

"health":<String_value>,

"clbkplanerxrate":<Double_value>,

"nodegroup_list":<String_value>,

"ipaddress":<String_value>,

"profile_name":<String_value>}]}
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_cluster **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_cluster **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_cluster resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_cluster **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_cluster resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_cluster **?count=yes** Use this query-parameter to get the count of ns_cluster resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_cluster":[{

"add_nodegroup":<Boolean_value>,

"mgmt_cpu_usage":<Double_value>,

"backplane":<String_value>,

"clusterid":<Integer_value>,

"nodeid":<Integer_value>,

"state":<String_value>,

"channel_interface_list":[{

"local_node":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"nodeid":<Integer_value>,
```

```
"id":<String_value>,

"parent_id":<String_value>}],

"totpropagationtimeout":<Double_value>,

"totsteeredpkts":<Double_value>,

"password":<String_value>,

"clag_intf_list":<String_value>,

"process_local":<Boolean_value>,

"act_id":<String_value>,

"numdfddroppkts":<Double_value>,

"clviewleader":<String_value>,

"inc_mode":<Boolean_value>,

"snip":<String_value>,

"clnumnodes":<Integer_value>,

"http_req":<Double_value>,

"scheduleId":<String_value>,

"sync_operation":<Boolean_value>,

"cpu_usage":<Double_value>,

"clbkplanerx":<Double_value>,

"clnodes":<String_value>,

"clbkplanetx":<Double_value>,

"preemption":<Boolean_value>,

"iscco":<Boolean_value>,
```

```
"clip":<String_value>,

"operationalstate":<String_value>,

"tx":<Double_value>,

"clcurstatus":<String_value>,

"rx":<Double_value>,

"memory_usage":<Double_value>,

"node_group":<String_value>,

"clbkplanetxrate":<Double_value>,

"health":<String_value>,

"clbkplanerxrate":<Double_value>,

"nodegroup_list":<String_value>,

"ipaddress":<String_value>,

"profile_name":<String_value>}]}
```

MODIFY_PASSWORD

URL: https://<MGMT-IP>/nitro/v2/config/ns_cluster?action=modify_password;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_cluster": {

<b>"ipaddress":<String_value></b>,

"clusterid":<Integer_value>,

"password":<String_value>,

"process_local":<Boolean_value>,

"clip":<String_value>,

"node_group":<String_value>,

"nodeid":<Integer_value>,

"snip":<String_value>,

"preemption":<Boolean_value>,

"iscco":<Boolean_value>,

"nodegroup_list":<String_value[]>,

"clnodes":<String_value>,

"profile_name":<String_value>,

"add_nodegroup":<Boolean_value>,

"backplane":<String_value>,

"state":<String_value>,

"inc_mode":<Boolean_value>,

"scheduleId":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_cluster":[{

"add_nodegroup":<Boolean_value>,

"mgmt_cpu_usage":<Double_value>,

"backplane":<String_value>,

"clusterid":<Integer_value>,

"nodeid":<Integer_value>,

"state":<String_value>,

"channel_interface_list":[{

"local_node":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"nodeid":<Integer_value>,

"id":<String_value>,

"parent_id":<String_value>}],

"totpropagationtimeout":<Double_value>,

"totsteeredpkts":<Double_value>,

"password":<String_value>,

"clag_intf_list":<String_value>,

"process_local":<Boolean_value>,

"act_id":<String_value>,
```

```
"numdfddroppkts":<Double_value>,

"clviewleader":<String_value>,

"inc_mode":<Boolean_value>,

"snip":<String_value>,

"clnumnodes":<Integer_value>,

"http_req":<Double_value>,

"scheduleId":<String_value>,

"sync_operation":<Boolean_value>,

"cpu_usage":<Double_value>,

"clbkplanerx":<Double_value>,

"clnodes":<String_value>,

"clbkplanetx":<Double_value>,

"preemption":<Boolean_value>,

"iscco":<Boolean_value>,

"clip":<String_value>,

"operationalstate":<String_value>,

"tx":<Double_value>,

"clcurstatus":<String_value>,

"rx":<Double_value>,

"memory_usage":<Double_value>,

"node_group":<String_value>,

"clbkplanetxrate":<Double_value>,
```

```
"health":<String_value>,

"clbkplanerxrate":<Double_value>,

"nodegroup_list":<String_value>,

"ipaddress":<String_value>,

"profile_name":<String_value>}]}
```

FORM_CLUSTER

URL: https://<MGMT-IP>/nitro/v2/config/ns_cluster?action=form_cluster;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_cluster": {

<b>"ipaddress":<String_value></b>,

"clusterid":<Integer_value>,

"password":<String_value>,

"process_local":<Boolean_value>,

"clip":<String_value>,

"node_group":<String_value>,

"nodeid":<Integer_value>,
```

```
"snip":<String_value>,

"preemption":<Boolean_value>,

"iscco":<Boolean_value>,

"nodegroup_list":<String_value[]>,

"clnodes":<String_value>,

"profile_name":<String_value>,

"add_nodegroup":<Boolean_value>,

"backplane":<String_value>,

"state":<String_value>,

"inc_mode":<Boolean_value>,

"scheduleId":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_cluster":[{

"add_nodegroup":<Boolean_value>,

"mgmt_cpu_usage":<Double_value>,

"backplane":<String_value>,

"clusterid":<Integer_value>,

"nodeid":<Integer_value>,
```

```
"state":<String_value>,

"channel_interface_list":[{

"local_node":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"nodeid":<Integer_value>,

"id":<String_value>,

"parent_id":<String_value>}],

"totpropagationtimeout":<Double_value>,

"totsteeredpkts":<Double_value>,

"password":<String_value>,

"clag_intf_list":<String_value>,

"process_local":<Boolean_value>,

"act_id":<String_value>,

"numdfddroppkts":<Double_value>,

"clviewleader":<String_value>,

"inc_mode":<Boolean_value>,

"snip":<String_value>,

"clnumnodes":<Integer_value>,

"http_req":<Double_value>,

"scheduleId":<String_value>,

"sync_operation":<Boolean_value>,
```

```
"cpu_usage":<Double_value>,

"clbkplanerx":<Double_value>,

"clnodes":<String_value>,

"clbkplanetx":<Double_value>,

"preemption":<Boolean_value>,

"iscco":<Boolean_value>,

"clip":<String_value>,

"operationalstate":<String_value>,

"tx":<Double_value>,

"clcurstatus":<String_value>,

"rx":<Double_value>,

"memory_usage":<Double_value>,

"node_group":<String_value>,

"clbkplanetxrate":<Double_value>,

"health":<String_value>,

"clbkplanerxrate":<Double_value>,

"nodegroup_list":<String_value>,

"ipaddress":<String_value>,

"profile_name":<String_value>}]}
```

JOIN

URL: https://<MGMT-IP>/nitro/v2/config/ns_cluster?action=join;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_cluster": {

<b>"ipaddress":<String_value></b>,

"clusterid":<Integer_value>,

"password":<String_value>,

"process_local":<Boolean_value>,

"clip":<String_value>,

"node_group":<String_value>,

"nodeid":<Integer_value>,

"snip":<String_value>,

"preemption":<Boolean_value>,

"iscco":<Boolean_value>,

"nodegroup_list":<String_value[]>,

"clnodes":<String_value>,

"profile_name":<String_value>,

"add_nodegroup":<Boolean_value>,

"backplane":<String_value>,
```

```
"state":<String_value>,

"inc_mode":<Boolean_value>,

"scheduleId":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_cluster":[{

"add_nodegroup":<Boolean_value>,

"mgmt_cpu_usage":<Double_value>,

"backplane":<String_value>,

"clusterid":<Integer_value>,

"nodeid":<Integer_value>,

"state":<String_value>,

"channel_interface_list":[{

"local_node":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"nodeid":<Integer_value>,

"id":<String_value>,

"parent_id":<String_value>}],
```

```
"totpropagationtimeout":<Double_value>,

"totsteeredpkts":<Double_value>,

"password":<String_value>,

"clag_intf_list":<String_value>,

"process_local":<Boolean_value>,

"act_id":<String_value>,

"numdfddroppkts":<Double_value>,

"clviewleader":<String_value>,

"inc_mode":<Boolean_value>,

"snip":<String_value>,

"clnumnodes":<Integer_value>,

"http_req":<Double_value>,

"scheduleId":<String_value>,

"sync_operation":<Boolean_value>,

"cpu_usage":<Double_value>,

"clbkplanerx":<Double_value>,

"clnodes":<String_value>,

"clbkplanetx":<Double_value>,

"preemption":<Boolean_value>,

"iscco":<Boolean_value>,

"clip":<String_value>,

"operationalstate":<String_value>,
```

```
"tx":<Double_value>,

"clcurstatus":<String_value>,

"rx":<Double_value>,

"memory_usage":<Double_value>,

"node_group":<String_value>,

"clbkplanetxrate":<Double_value>,

"health":<String_value>,

"clbkplanerxrate":<Double_value>,

"nodegroup_list":<String_value>,

"ipaddress":<String_value>,

"profile_name":<String_value>}]}
```

ns_cluster_nodegroup

Configuration for ns_cluster_nodegroup resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
password	<String>	Read-write	Password of the Configuration Coordinator node(Cluster IP). Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
clip	<String>	Read-write	Cluster IPAddress where this nodegroup will be added. Minimum length = 1 Maximum length = 64
name	<String>	Read-write	Cluster Node Group Name.

Operations

(click to see Properties)

- ADD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns_cluster_nodegroup?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_cluster_nodegroup": {

"password":<String_value>,

"name":<String_value>,

"clip":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_cluster_nodegroup":[{

"password":<String_value>,

"clip":<String_value>,

"name":<String_value>}]}
```

ns_clusternode

Configuration for Cluster node information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
clusterid	<Integer>	Read-write	Cluster Instance ID. Minimum value = 1 Maximum value =
nodeid	<Integer>	Read-write	Cluster Node ID. Maximum value =
clnodeip	<String>	Read-write	Cluster Node IP. Minimum length = 1 Maximum length = 64
password	<String>	Read-write	Password of the CCO node(CLIP). Minimum length = 1 Maximum length = 128
iscco	<Boolean>	Read-write	Is CCO.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
clip	<String>	Read-write	Cluster IP. Minimum length = 1 Maximum length = 64
node_group	<String>	Read-write	Node Group where the node belongs in L3 Cluster.
islocalsdx	<String>	Read-write	Is Node Belonging to Local SDX.
clnodehealth	<String>	Read-only	Health of the node in the cluster.
nnmcurconn	<Double>	Read-only	Number of Connections open for node to node connection..
nnmerrmsend	<Double>	Read-only	Number of errors in sending node-to-node multicast/ broadcast messages.
nnmtotconntx	<Double>	Read-only	Number of node-to-node messages sent.
act_id	<String>	Read-only	Activity Id.
clnodeeffectivehealth	<String>	Read-only	Effective Health of the node in the cluster.
clptptx	<Double>	Read-only	Number of PTP packets transmitted by the node.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
clmasterstate	<String>	Read-only	Cluster Master State.
clsyncstate	<String>	Read-only	Sync state of the cluster node.
cltothbtx	<Double>	Read-only	Cluster Number of Heartbeats Sent.
nnmtotconnrx	<Double>	Read-only	Number of node-to-node messages received.
clptprx	<Double>	Read-only	Number of PTP packets received on the node.
clptpstate	<String>	Read-only	PTP state of the node. This state is Master for one node and Slave for the rest.
cltothbrx	<Double>	Read-only	Cluster Number of Heartbeats Received.
masterstate	<String>	Read-only	Master State of the clusternode.

Operations

(click to see Properties)

- REMOVE_NODE
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

REMOVE_NODE

URL: https://<MGMT-IP>/nitro/v2/config/ns_clusternode?action=remove_node;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_clusternode": {

<b>"nodeid":<Integer_value></b>,

"clusterid":<Integer_value>,

"password":<String_value>,

"islocalsdx":<String_value>,

"clnodeip":<String_value>,

"clip":<String_value>,

"node_group":<String_value>,

"iscco":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_clusternode":[{

"clnodehealth":<String_value>,

"clusterid":<Integer_value>,

"nodeid":<Integer_value>,

"nnmcurconn":<Double_value>,

"clnodeip":<String_value>,

"nnmerrmsend":<Double_value>,

"nnmtotconntx":<Double_value>,
```

```
"password":<String_value>,

"act_id":<String_value>,

"iscco":<Boolean_value>,

"clnodeeffectivehealth":<String_value>,

"clip":<String_value>,

"clptptx":<Double_value>,

"clmasterstate":<String_value>,

"clsyncstate":<String_value>,

"cltothbtx":<Double_value>,

"nnmtotconnrx":<Double_value>,

"clptprx":<Double_value>,

"clptpstate":<String_value>,

"node_group":<String_value>,

"cltothbrx":<Double_value>,

"islocalsdx":<String_value>,

"masterstate":<String_value>}]}
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_clusternode **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_clusternode **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_clusternode resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_clusternode **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_clusternode resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_clusternode **?count=yes** Use this query-parameter to get the count of ns_clusternode resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_clusternode":[{

"clnodehealth":<String_value>,

"clusterid":<Integer_value>,

"nodeid":<Integer_value>,

"nnmcurconn":<Double_value>,

"clnodeip":<String_value>,

"nnmerrmsend":<Double_value>,
```

```
"nnmtotconntx":<Double_value>,

"password":<String_value>,

"act_id":<String_value>,

"iscco":<Boolean_value>,

"clnodeeffectivehealth":<String_value>,

"clip":<String_value>,

"clptptx":<Double_value>,

"clmasterstate":<String_value>,

"clsyncstate":<String_value>,

"cltothbtx":<Double_value>,

"nnmtotconnrx":<Double_value>,

"clptprx":<Double_value>,

"clptpstate":<String_value>,

"node_group":<String_value>,

"cltothbrx":<Double_value>,

"islocalsdx":<String_value>,

"masterstate":<String_value>}]}
```

ns_command

Configuration for Citrix ADC configuration template command resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_name	<String>	Read-write	.
id	<String>	Read-write	Id is system generated key.
command	<String>	Read-write	Citrix ADC Command.
parent_id	<String>	Read-write	.
index	<Integer>	Read-only	Command Index.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ns_config_replicate

Configuration for Citrix ADC Configuration Replicate resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
source_device	<String>	Read-write	Source Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
target_device	<String[]>	Read-write	Target device IP Address Array on which replicate config from source device.

Operations

(click to see Properties)

- REPLICATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

REPLICATE

URL: https://<MGMT-IP>/nitro/v2/config/ns_config_replicate?action=replicate;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_config_replicate": {

"target_device":<String_value[]>,

"source_device":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_config_replicate":[{

"source_device":<String_value>,

"target_device":<String_value>}]}
```

ns_detail_vlan

Configuration for Citrix ADC Detail Vlan resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_vlan_ip_address	<String>	Read-write	VPX IP Address. Minimum length = 1 Maximum length = 64
network_interfaces	<network_interface[]>	Read-write	Network Interfaces.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_ipv6_routing	<String>	Read-write	VLAN for Network Interface on VM Instance.
ns_vlan_id	<Integer>	Read-write	VLAN Id. Maximum value =

Operations

(click to see Properties)

- ADD
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns_detail_vlan?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_detail_vlan": {

"ns_vlan_ip_address":<String_value>,

"ns_vlan_id":<Integer_value>,

"if_ipv6_routing":<String_value>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"is_member_ifc":<Boolean_value>,
```

```
"id":<String_value>,

"l2_enabled":<Boolean_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv4":<String_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"interface_name":<String_value>,

"mac_mode":<String_value>,

"vlan":<Integer_value>,

"managed_device_id":<String_value>,

"vlan_whitelist":<String_value>}]}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_detail_vlan":[{

"ns_vlan_ip_address":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,
```

```
"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"if_ipv6_routing":<String_value>,

"ns_vlan_id":<Integer_value>}]}
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_detail_vlan **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_detail_vlan **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_detail_vlan resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_detail_vlan **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_detail_vlan resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_detail_vlan **?count=yes** Use this query-parameter to get the count of ns_detail_vlan resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_detail_vlan":[{

"ns_vlan_ip_address":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,
```

```
"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,
```

```
"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"if_ipv6_routing":<String_value>,

"ns_vlan_id":<Integer_value>}]}
```

ns_device_profile

Configuration for Device profile for Citrix ADC(MPX/VPX/CPX/Gateway) instances resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Profile Name. Minimum length = 1 Maximum length = 128
svm_ns_comm	<String>	Read-write	Communication protocol (http or https) with Instances. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 10
use_global_setting_for_communication_with_ns	<Boolean>	Read-write	True, if the communication with Instance needs to be global and not device specific.
id	<String>	Read-write	Id is system generated key for all the device profiles.
type	<String>	Read-write	Profile Type, This must be with in specified supported instance types: blx,sdvanvw,ns,nssdx,cbwanopt,cpx. Minimum length = 1 Maximum length = 128
ns_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix ADC profiles.
password	<String>	Read-write	Instance credentials.Password for this profile. Minimum length = 1 Maximum length = 127
snmpsecuritylevel	<String>	Read-write	SNMP v3 security level for this profile.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
username	<String>	Read-write	Instance credentials.Username provided in the profile will be used to contact the instance. Minimum length = 1 Maximum length = 127
snmpauthprotocol	<String>	Read-write	SNMP v3 auth protocol for this profile.
ssh_port	<String>	Read-write	SSH port to connect to the device.
snmpprivprotocol	<String>	Read-write	SNMP v3 priv protocol for this profile.
host_password	<String>	Read-write	Host Password for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpversion	<String>	Read-write	SNMP version for this profile.
passphrase	<String>	Read-write	Passphrase with which private key is encrypted.
host_username	<String>	Read-write	Host User Name for this profile.Used for BLX form factor of ADC.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 127
snmpsecurityname	<String>	Read-write	SNMP v3 security name for this profile. Maximum length = 31
ssl_private_key	<String>	Read-write	SSL Private Key for key based authentication.
ssl_cert	<String>	Read-write	SSL Certificate for certificate based authentication.
http_port	<Integer>	Read-write	HTTP port to connect to the device.
snmpcommunity	<String>	Read-write	SNMP community for this profile. Maximum length = 31
https_port	<Integer>	Read-write	HTTPS port to connect to the device.
max_wait_time_reboot	<String>	Read-write	Max waiting time to reboot Citrix ADC.
snmpprivpassword	<String>	Read-write	SNMP v3 priv password for this profile. Minimum length = 8 Maximum length = 31

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cb_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix SD-WAN profiles.
snmpauthpassword	<String>	Read-write	SNMP v3 auth password for this profile. Minimum length = 8 Maximum length = 31
is_default	<Boolean>	Read-only	True, if this profile is system generated and can not be deleted.
tenant_id	<String>	Read-only	Tenant Id of device profile.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns_device_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_device_profile": {

<b>"name":<String_value></b>,

<b>"password":<String_value></b>,

<b>"username":<String_value></b>,

"snmpsecuritylevel":<String_value>,

"svm_ns_comm":<String_value>,

"ssh_port":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"type":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"ns_profile_name":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,
```

```
"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"tenant_id":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,
```

```
"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/ns_device_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_device_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_device_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_device_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_device_profile **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_device_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_device_profile **?count=yes** Use this query-parameter to get the count of ns_device_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"tenant_id":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,
```

```
"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_device_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"tenant_id":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,
```

```
"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/ns_device_profile/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_device_profile":{

<b>"id":<String_value></b>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"name":<String_value>,
```

```
"svm_ns_comm":<String_value>,

"ssh_port":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"type":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"ns_profile_name":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"tenant_id":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,
```

```
"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

ns_emon_poll_policy

Configuration for Citrix ADC EMON Polling Policy resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
polling_interval	<Integer>	Read-write	Frequency of polling in minutes. Minimum value = 5 Maximum value =
ns_ip_address	<String[]>	Read-write	List of Citrix ADC IP Address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
act_id	<String>	Read-write	Activity ID that is used by GUI to track the polling status..

Operations

(click to see Properties)

- DO_POLL
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

DO_POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_emon_poll_policy?action=do_poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_emon_poll_policy": {

<b>"polling_interval":<Integer_value></b>,

"ns_ip_address":<String_value[]>,

"act_id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_emon_poll_policy":[{

"polling_interval":<Integer_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_emon_poll_policy **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_emon_poll_policy":[{

"polling_interval":<Integer_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/ns_emon_poll_policy/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_emon_poll_policy":{

<b>"polling_interval":<Integer_value></b>,

"ns_ip_address":<String_value[]>,

"act_id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_emon_poll_policy":[{

"polling_interval":<Integer_value>,

"ns_ip_address":<String_value>,

"act_id":<String_value>}]}
```

ns_filterpolicy_template

Configuration for Citrix ADC Filter Policy Configure Task resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
priority	<Integer>	Read-write	Policy Priority.
device_groups	<String[]>	Read-write	Device Group Array on which for which template is applied.
name	<String>	Read-write	Name of Policy.
variables	<variable_values[]>	Read-write	Values of Variables used in commands of the configuration template.
state	<String>	Read-write	State of the Policy: enabled/disabled.
devices	<String[]>	Read-write	Device IP Address Array on which template is applied.
actionType	<String>	Read-write	Action Type of the Policy.
actionName	<String>	Read-write	Action Name of the Policy.
expression	<String>	Read-write	Expression of the Policy.

Operations

(click to see Properties)

- ADD
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns_filterpolicy_template?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_filterpolicy_template": {

"name":<String_value>,

"devices":<String_value[]>,

"actionType":<String_value>,

"device_groups":<String_value[]>,

"actionName":<String_value>,

"variables":[{

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"values_enum_db":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"description":<String_value>,

"type":<String_value>,

"valuex":<String_value>,

"device_values":[{

"parent_name":<String_value>,
```

```
"value":<String_value>,

"id":<String_value>,

"device":<String_value>,

"device_group":<String_value>,

"valuex":<String_value>,

"parent_id":<String_value>}],

"display_name":<String_value>}],

"priority":<Integer_value>,

"state":<String_value>,

"expression":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_filterpolicy_template":[{

"priority":<Integer_value>,

"device_groups":<String_value>,

"name":<String_value>,

"variables":[{

"device_values_mapping":<String_value>,

"parent_name":<String_value>,
```

```
"value":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"valuex":<String_value>,

"device_group":<String_value>,

"device":<String_value>,

"parent_id":<String_value>}],

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>}],

"state":<String_value>,

"devices":<String_value>,

"actionType":<String_value>,

"actionName":<String_value>,
```

```
"expression":<String_value>}]}
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_filterpolicy_template **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_filterpolicy_template **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_filterpolicy_template resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_filterpolicy_template **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_filterpolicy_template resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_filterpolicy_template **?count=yes** Use this query-parameter to get the count of ns_filterpolicy_template resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_filterpolicy_template":[{

"priority":<Integer_value>,

"device_groups":<String_value>,

"name":<String_value>,

"variables":[{

"device_values_mapping":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"valuex":<String_value>,

"device_group":<String_value>,

"device":<String_value>,

"parent_id":<String_value>}],

"parent_id":<String_value>,
```

```
"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>}],

"state":<String_value>,

"devices":<String_value>,

"actionType":<String_value>,

"actionName":<String_value>,

"expression":<String_value>}]}
```

ns_hapair_template

Configuration for Citrix ADC HA Pair Configure Template resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
nssecondary_ip_address	<String>	Read-write	Secondary HA Node NSIP Address. Minimum length = 1 Maximum length = 64
backplane	<String>	Read-write	Backplane Interface. Minimum length = 1 Maximum length = 32
clusterid	<Integer>	Read-write	Cluster Id.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum value = 1 Maximum value =
secondary_nodeid	<Integer>	Read-write	Secondary Node ID. Maximum value =
slack_profile	<String>	Read-write	Slack profile.
execute_sequentially	<Boolean>	Read-write	True if configuration template is to be applied to devices sequentially.
nsprimary_ip_address	<String>	Read-write	Primary HA Node NSIP Address. Minimum length = 1 Maximum length = 64
clip	<String>	Read-write	Cluster IPAddress. Minimum length = 1 Maximum length = 64
primary_ip_address	<String>	Read-write	Primary HA Node IP Address. Minimum length = 1 Maximum length = 64
configCoordinator	<String>	Read-write	Node to be designated as Config Coordinator - primary or secondary.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
scheduleTimesEpoch	<String>	Read-write	Schedule time epoch (string representation of 11 digit numbers)..
secondary_ip_address	<String>	Read-write	Secondary HA Node IP Address. Minimum length = 1 Maximum length = 64
name	<String>	Read-write	Name. Maximum length = 128
primary_nodeid	<Integer>	Read-write	Prmiary Node ID. Maximum value =
on_error	<String>	Read-write	Behaviour on encountering error while applying configuration template on a device: CONTINUE| EXIT. Minimum length = 1 Maximum length = 16
inc_enabled	<Boolean>	Read-write	Turn on INC mode on self node.
scheduleId	<String>	Read-write	scheduleId is used to refer maintenaince schedule task.
mail_profiles	<String>	Read-write	Comma separated list of Mail profiles.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_hapair_template **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_hapair_template **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_hapair_template resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_hapair_template **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_hapair_template resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_hapair_template **?count=yes** Use this query-parameter to get the count of ns_hapair_template resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_hapair_template":[{

"nssecondary_ip_address":<String_value>,
```

```
"backplane":<String_value>,

"clusterid":<Integer_value>,

"secondary_nodeid":<Integer_value>,

"slack_profile":<String_value>,

"execute_sequentially":<Boolean_value>,

"nsprimary_ip_address":<String_value>,

"act_id":<String_value>,

"clip":<String_value>,

"primary_ip_address":<String_value>,

"configCoordinator":<String_value>,

"scheduleTimesEpoch":<String_value>,

"secondary_ip_address":<String_value>,

"name":<String_value>,

"primary_nodeid":<Integer_value>,

"on_error":<String_value>,

"inc_enabled":<Boolean_value>,

"scheduleId":<String_value>,

"mail_profiles":<String_value>,

"sync_operation":<Boolean_value>}]}
```

ns_image

Configuration for Citrix ADC Build File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/ns_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_image **?count=yes** Use this query-parameter to get the count of ns_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_image":[{
```

```
"file_location_path":<String_value>,

"file_name":<String_value>,

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"is_dir":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_image":[{

"file_location_path":<String_value>,

"file_name":<String_value>,
```

```
"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"is_dir":<Boolean_value>}]}
```

UPLOAD

URL: https://<MGMT-IP>/nitro/v2/upload/ns_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

ns_interface

Configuration for Citrix ADC Interface resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_cluster_node	<Boolean>	Read-write	Is the node is part of cluster or not.
vif_list	<String[]>	Read-write	Virtual Interface List.
pif_list	<String[]>	Read-write	Physical Interface List.
nodeid	<Integer>	Read-write	Node Id applicable on cluster. Maximum value =
ip_address	<String>	Read-write	Citrix ADC IP Address.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_interface **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_interface **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_interface resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_interface **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_interface resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_interface **?count=yes** Use this query-parameter to get the count of ns_interface resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_interface":[{

"is_cluster_node":<Boolean_value>,

"vif_list":<String_value>,

"pif_list":<String_value>,

"nodeid":<Integer_value>,

"ip_address":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_interface/ip_address_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_interface":[{

"is_cluster_node":<Boolean_value>,

"vif_list":<String_value>,

"pif_list":<String_value>,

"nodeid":<Integer_value>,

"ip_address":<String_value>}]}
```

ns_l7_latency_config

Configuration for L7 Latency configuration on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vserverType	<String>	Read-write	Virtual server type(lb/ sslvpn/CR). Minimum length = 2 Maximum length = 64
l7LatencyThresholdNotifyIntetval	<Integer>	Read-write	L7 Latency Threshold Notify Intetval.
l7LatencyNotifyCount	<Integer>	Read-write	L7 Latency Notify Count.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
l7LatencyThresholdWaitInterval	<Integer>	Read-write	L7 Latency Threshold Wait Interval.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
l7LatencyThresholdFactor	<Integer>	Read-write	L7 Latency Threshold Factor Value.
vserverName	<String>	Read-write	Virtual server name. Maximum length = 255
l7LatencyMonitoring	<String>	Read-write	ENABLE/DISABLE. Maximum length = 512
latencyName	<String>	Read-only	Name of latency profile.

Operations

(click to see Properties)

- GET (ALL)
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_l7_latency_config **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_l7_latency_config **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_l7_latency_config resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_l7_latency_config **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_l7_latency_config resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_l7_latency_config **?count=yes** Use this query-parameter to get the count of ns_l7_latency_config resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_l7_latency_config":[{

"vserverType":<String_value>,

"l7LatencyThresholdNotifyIntetval":<Integer_value>,

"l7LatencyNotifyCount":<Integer_value>,

"l7LatencyThresholdWaitInterval":<Integer_value>,

"ns_ip_address":<String_value>,

"l7LatencyThresholdFactor":<Integer_value>,

"latencyName":<String_value>,

"vserverName":<String_value>,

"l7LatencyMonitoring":<String_value>}]}
```

MODIFY

URL: https://<MGMT-IP>/nitro/v2/config/ns_l7_latency_config/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_l7_latency_config":{

"l7LatencyThresholdNotifyIntetval":<Integer_value>,

"ns_ip_address":<String_value>,

"vserverType":<String_value>,

"vserverName":<String_value>,

"l7LatencyThresholdWaitInterval":<Integer_value>,

"l7LatencyMonitoring":<String_value>,

"l7LatencyNotifyCount":<Integer_value>,

"l7LatencyThresholdFactor":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_l7_latency_config":[{
```

```
"vserverType":<String_value>,

"l7LatencyThresholdNotifyIntetval":<Integer_value>,

"l7LatencyNotifyCount":<Integer_value>,

"l7LatencyThresholdWaitInterval":<Integer_value>,

"ns_ip_address":<String_value>,

"l7LatencyThresholdFactor":<Integer_value>,

"latencyName":<String_value>,

"vserverName":<String_value>,

"l7LatencyMonitoring":<String_value>}]}
```

ns_mpxcluster_template

Configuration for Citrix ADC MPX Cluster Template resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cco	<String>	Read-write	CCO IP address.
clip	<String>	Read-write	Cluster IPAddress. Minimum length = 1 Maximum length = 64
backplane	<String>	Read-write	Backplane Interface. Minimum length = 1 Maximum length = 32

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
clusterid	<Integer>	Read-write	Cluster Id. Minimum value = 1 Maximum value =
devices	<String[]>	Read-write	Device IP Address Array on which template is applied.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_mpxcluster_template **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_mpxcluster_template **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_mpxcluster_template resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_mpxcluster_template **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_mpxcluster_template resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_mpxcluster_template **?count=yes** Use this query-parameter to get the count of ns_mpxcluster_template resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_mpxcluster_template":[{

"cco":<String_value>,

"sync_operation":<Boolean_value>,

"clip":<String_value>,

"backplane":<String_value>,

"clusterid":<Integer_value>,

"devices":<String_value>}]}
```

ns_network_interface

Configuration for NS Network Interfaces resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
devicename	<String>	Read-write	Device/Interface Name. Minimum length = 1 Maximum length = 100
id	<String>	Read-write	Id is system generated key.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
nicerrifoutdiscards	<Double>	Read-write	Number of error free outgoing packets discarded.
errifindiscards	<Double>	Read-write	Number of error free incoming packets discarded.
display_name	<String>	Read-write	Device Display Name. Minimum length = 1 Maximum length = 64
partition_name	<String>	Read-only	Partition Name.
poll_time	<Integer>	Read-only	Last Polling Time.
state	<String>	Read-only	State Enabled| Disabled.
hostname	<String>	Read-only	Assign hostname to managed device, if this is not provided, name will be set as host name .
description	<String>	Read-only	Interface description.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_network_interface **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_network_interface":[{

"partition_name":<String_value>,

"devicename":<String_value>,

"poll_time":<Integer_value>,

"id":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"nicerrifoutdiscards":<Double_value>,

"errifindiscards":<Double_value>,

"hostname":<String_value>,

"display_name":<String_value>,

"description":<String_value>}]}
```

ns_ns_appflowcollector

Configuration for AppFlow Collector for Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
transport	<String>	Read-write	Transport type like logstream, IPfix. Minimum length = 1 Maximum length = 20
name	<String>	Read-write	name. Minimum length = 1 Maximum length = 256
port	<Integer>	Read-write	port.
state	<String>	Read-write	state of the collector. Minimum length = 1 Maximum length = 10
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
ipaddress	<String>	Read-write	ipaddress. Minimum length = 1 Maximum length = 20
poll_time	<Integer>	Read-only	Last Polling Time.
display_name	<String>	Read-only	Display Name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_ns_appflowcollector **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_ns_appflowcollector **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_ns_appflowcollector resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_ns_appflowcollector **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_ns_appflowcollector resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_ns_appflowcollector **?count=yes** Use this query-parameter to get the count of ns_ns_appflowcollector resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ns_appflowcollector":[{

"transport":<String_value>,
```

```
"name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"port":<Integer_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"id":<String_value>,

"ipaddress":<String_value>}]}
```

ns_ns_appflowparam

Configuration for Appflow param for Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
httpquerywithurl	<String>	Read-write	httpquerywithurl Enabled/Disabled. Minimum length = 1 Maximum length = 10
urlcategory	<String>	Read-write	URL Category.
aaausername	<String>	Read-write	AAA Username logging.
template_interval	<Integer>	Read-write	Template refresh interval.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
httpxforwardedfor	<String>	Read-write	. Minimum length = 1 Maximum length = 10
httpmethod	<String>	Read-write	HTTP Method logging.
observationdomainName	<String>	Read-write	observationdomainName. Minimum length = 1 Maximum length = 1024
securityinsighttraffic	<String>	Read-write	Security Insight Traffic logging.
httpcookie	<String>	Read-write	httpCookie Enabled/ Disabled. Minimum length = 1 Maximum length = 10
events	<String>	Read-write	Events Enabled/ Disabled.
httpdomain	<String>	Read-write	HTTP Domain Traffic logging.
metrics	<String>	Read-write	Metrics Enabled/ Disabled.
auditlogs	<String>	Read-write	Auditlogs Enabled/ Disabled.
observationdomainid	<String>	Read-write	observationdomainid.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 256
httpuseragent	<String>	Read-write	HTTP User-Agent logging.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_ns_appflowparam **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_ns_appflowparam **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_ns_appflowparam resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_ns_appflowparam **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_ns_appflowparam resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_ns_appflowparam **?count=yes** Use this query-parameter to get the count of ns_ns_appflowparam resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ns_appflowparam":[{

"httpquerywithurl":<String_value>,

"urlcategory":<String_value>,

"aaausername":<String_value>,

"template_interval":<Integer_value>,

"httpxforwardedfor":<String_value>,

"httpmethod":<String_value>,

"observationdomainName":<String_value>,

"securityinsighttraffic":<String_value>,

"httpcookie":<String_value>,

"events":<String_value>,

"httpdomain":<String_value>,

"metrics":<String_value>,

"auditlogs":<String_value>,

"observationdomainid":<String_value>,

"httpuseragent":<String_value>}]}
```

ns_ns_ip

Configuration for IP on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
save_config	<Boolean>	Read-write	true, if save config is required.
type	<String>	Read-write	Type of IP Address, Possible Values: SNIP. Minimum length = 1 Maximum length = 64
netmask	<String>	Read-write	netmask. Minimum length = 1 Maximum length = 64
ipaddress	<String>	Read-write	Citrix ADC ipaddress that needs to be created. Minimum length = 1 Maximum length = 64
ns_ip_address	<String>	Read-write	Citrix ADC IP Address where IP needs to be created. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- ADD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns_ns_ip?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ns_ip": {

<b>"type":<String_value></b>,

<b>"netmask":<String_value></b>,

<b>"ipaddress":<String_value></b>,

<b>"ns_ip_address":<String_value></b>,

"save_config":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ns_ip":[{

"save_config":<Boolean_value>,

"type":<String_value>,

"netmask":<String_value>,

"ipaddress":<String_value>,

"ns_ip_address":<String_value>}]}
```

ns_ns_ipv6

Configuration for IP on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
type	<String>	Read-write	Type of IPv6 Address.
scope	<String>	Read-write	Scope of IPv6 Address, Possible Values: GLobal and Link Local.
ipv6address	<String>	Read-write	Citrix ADC ipaddress that needs to be created. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ns_ns_mode

Configuration for Citrix ADC Mode resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
rise_rhi	<Boolean>	Read-write	true, if rise_rhi feature is enabled.
sradv	<Boolean>	Read-write	true, if sradv feature is enabled.
pmtud	<Boolean>	Read-write	true, if pmtud feature is enabled.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
dradv	<Boolean>	Read-write	true, if dradv feature is enabled.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
tcpb	<Boolean>	Read-write	true, if tcpb feature is enabled.
fr	<Boolean>	Read-write	true, if fr feature is enabled.
dradv6	<Boolean>	Read-write	true, if dradv6 feature is enabled.
edge	<Boolean>	Read-write	true, if edge feature is enabled.
usnip	<Boolean>	Read-write	true, if usnip feature is enabled.
bridgebpdus	<Boolean>	Read-write	true, if bridgebpdus feature is enabled.
cka	<Boolean>	Read-write	true, if cka feature is enabled.
ulfd	<Boolean>	Read-write	true, if ulfd feature is enabled.
rise_apbr	<Boolean>	Read-write	true, if rise_apbr feature is enabled.
sradv6	<Boolean>	Read-write	true, if sradv6 feature is enabled.
ns_ip_address_arr	<String[]>	Read-write	List of Citrix ADC IP Addresses. Minimum length = 1 Maximum length = 64
mbf	<Boolean>	Read-write	true, if mbf feature is enabled.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
l3	<Boolean>	Read-write	true, if l3 feature is enabled.
l2	<Boolean>	Read-write	true, if l2 feature is enabled.
mediaclassification	<Boolean>	Read-write	true, if mediaclassification feature is enabled.
iradv	<Boolean>	Read-write	true, if iradv feature is enabled.
usip	<Boolean>	Read-write	true, if usip feature is enabled.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_ns_mode **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_ns_mode **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_ns_mode resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_ns_mode **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_ns_mode resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_ns_mode **?count=yes** Use this query-parameter to get the count of ns_ns_mode resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ns_mode":[{

"rise_rhi":<Boolean_value>,

"sradv":<Boolean_value>,

"pmtud":<Boolean_value>,

"ns_ip_address":<String_value>,

"dradv":<Boolean_value>,

"tcpb":<Boolean_value>,

"fr":<Boolean_value>,

"dradv6":<Boolean_value>,

"edge":<Boolean_value>,

"usnip":<Boolean_value>,

"bridgebpdus":<Boolean_value>,

"cka":<Boolean_value>,

"ulfd":<Boolean_value>,

"rise_apbr":<Boolean_value>,

"sradv6":<Boolean_value>,
```

```
"ns_ip_address_arr":<String_value>,

"mbf":<Boolean_value>,

"l3":<Boolean_value>,

"l2":<Boolean_value>,

"mediaclassification":<Boolean_value>,

"iradv":<Boolean_value>,

"usip":<Boolean_value>}]}
```

ns_ns_rollbackcmd

Configuration for Citrix ADC Rollback command resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
rollback_command	<String>	Read-write	roll back command.
response	<String>	Read-write	response.
command	<String>	Read-write	command .
ns_ip_address	<String>	Read-write	Citrix ADC IP Address.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_ns_rollbackcmd **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_ns_rollbackcmd **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_ns_rollbackcmd resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_ns_rollbackcmd **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_ns_rollbackcmd resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_ns_rollbackcmd **?count=yes** Use this query-parameter to get the count of ns_ns_rollbackcmd resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ns_rollbackcmd":[{

"rollback_command":<String_value>,

"response":<String_value>,

"command":<String_value>,

"ns_ip_address":<String_value>}]}
```

ns_ns_runningconfig

Configuration for Citrix ADC Running Configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
partition_name	<String>	Read-only	Partition Name.
response	<String>	Read-only	response.
withdefaults	<String>	Read-only	withdefaults.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_ns_runningconfig **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ns_runningconfig":[{

"partition_name":<String_value>,

"response":<String_value>,
```

```
"ns_ip_address":<String_value>,

"withdefaults":<String_value>}]}
```

ns_ns_savedconfig

Configuration for Citrix ADC Saved Configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
response	<String>	Read-only	response.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_ns_savedconfig **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ns_savedconfig":[{

"response":<String_value>,

"ns_ip_address":<String_value>}]}
```

ns_ns_timeout

Configuration for Citrix ADC Timeout on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
timeout	<Integer>	Read-write	Timeout. Maximum value =
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- GET (ALL)
- SET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_ns_timeout **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_ns_timeout **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_ns_timeout resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_ns_timeout **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_ns_timeout resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_ns_timeout **?count=yes** Use this query-parameter to get the count of ns_ns_timeout resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ns_timeout":[{

"timeout":<Integer_value>,

"ns_ip_address":<String_value>}]}
```

SET

URL: https://<MGMT-IP>/nitro/v2/config/ns_ns_timeout/ns_ip_address_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_ns_timeout":{

"ns_ip_address":<String_value>,

"timeout":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_ns_timeout":[{

"timeout":<Integer_value>,

"ns_ip_address":<String_value>}]}
```

ns_partition

Configuration for Citrix ADC Partition resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
partitionmac	<String>	Read-only	vmac used by the partition.
partition_name	<String>	Read-only	Partition Name.
partition_id	<String>	Read-only	Partition ID.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_partition **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_partition":[{

"partitionmac":<String_value>,

"partition_name":<String_value>,

"id":<String_value>,

"partition_id":<String_value>,

"ns_ip_address":<String_value>}]}
```

ns_save_config

Configuration for Save configuration on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address_arr	<String[]>	Read-write	List of Citrix ADC IP Address. Minimum length = 1 Maximum length = 1024

Operations

(click to see Properties)

- ADD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns_save_config?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_save_config": {

<b>"ns_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_save_config":[{
```

```
"ns_ip_address_arr":<String_value>}]}
```

ns_snmp_config

Configuration for SNMP configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
save_config	<Boolean>	Read-write	true, if save config is required.
trapdestination	<String>	Read-write	State of configuration (enable/disable).
snmpauthprotocol	<String>	Read-write	SNMP v3 auth protocol.
snmpsecurityname	<String>	Read-write	SNMP v3 security name. Maximum length = 31
snmpprivprotocol	<String>	Read-write	SNMP v3 priv protocol.
ns_ip_address_arr	<String[]>	Read-write	List of Citrix ADC IP Address. Maximum length = 64
state	<String>	Read-write	State of configuration (enable/disable).
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
snmpsecuritylevel	<String>	Read-write	SNMP v3 security level.
snmpcommunity	<String>	Read-write	SNMP community. Maximum length = 31
snmpprivpassword	<String>	Read-write	SNMP v3 priv password. Minimum length = 8 Maximum length = 31
snmpversion	<String>	Read-write	SNMP version.
snmpauthpassword	<String>	Read-write	SNMP v3 auth password. Minimum length = 8 Maximum length = 31

Operations

(click to see Properties)

- ADD
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns_snmp_config?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_snmp_config": {

<b>"ns_ip_address":<String_value></b>,

<b>"snmpversion":<String_value></b>,

"snmpsecuritylevel":<String_value>,

"save_config":<Boolean_value>,

"ns_ip_address_arr":<String_value[]>,

"trapdestination":<String_value>,

"snmpprivpassword":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,

"state":<String_value>,

"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_snmp_config":[{

"save_config":<Boolean_value>,

"trapdestination":<String_value>,
```

```
"snmpauthprotocol":<String_value>,

"snmpsecurityname":<String_value>,

"snmpprivprotocol":<String_value>,

"ns_ip_address_arr":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"snmpsecuritylevel":<String_value>,

"snmpcommunity":<String_value>,

"snmpprivpassword":<String_value>,

"snmpversion":<String_value>,

"snmpauthpassword":<String_value>}]}
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/ns_snmp_config/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_snmp_config":{

"ns_ip_address":<String_value>,

"snmpsecuritylevel":<String_value>,
```

```
"save_config":<Boolean_value>,

"snmpversion":<String_value>,

"ns_ip_address_arr":<String_value[]>,

"trapdestination":<String_value>,

"snmpprivpassword":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,

"state":<String_value>,

"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_snmp_config":[{

"save_config":<Boolean_value>,

"trapdestination":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpsecurityname":<String_value>,

"snmpprivprotocol":<String_value>,
```

```
"ns_ip_address_arr":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"snmpsecuritylevel":<String_value>,

"snmpcommunity":<String_value>,

"snmpprivpassword":<String_value>,

"snmpversion":<String_value>,

"snmpauthpassword":<String_value>}]}
```

ns_sslcipher_binding

Configuration for Citrix ADC SSL Cipher Binding resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ciphername	<String>	Read-write	ciphername.
ciphergroupname	<String>	Read-write	ciphergroupname.
cipherpriority	<Integer>	Read-write	cipherpriority.
id	<String>	Read-write	Id is system generated key.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
partition_name	<String>	Read-only	Partition Name.
poll_time	<Integer>	Read-only	Last Polling Time.
display_name	<String>	Read-only	Display Name.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_sslcipher_binding **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_sslcipher_binding":[{

"ciphername":<String_value>,

"ciphergroupname":<String_value>,

"cipherpriority":<Integer_value>,

"partition_name":<String_value>,

"poll_time":<Integer_value>,

"id":<String_value>,

"display_name":<String_value>,
```

```
"ns_ip_address":<String_value>}]}
```

ns_sslparameter

Configuration for Citrix ADC SSL Parameter resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
denysslreneg	<String>	Read-write	denysslreneg.
insertionencoding	<String>	Read-write	insertionencoding.
defaultprofile	<Boolean>	Read-write	defaultprofile.
dropreqwithnohostheader	<String>	Read-write	dropreqwithnohostheader.
strictcachecks	<String>	Read-write	strictcachecks.
pushenctriggertimeout	<Integer>	Read-write	pushenctriggertimeout.
quantumsize	<Integer>	Read-write	quantumsize.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
ssltriggertimeout	<Integer>	Read-write	ssltriggertimeout.
pushflag	<Integer>	Read-write	pushflag.
id	<String>	Read-write	Id is system generated key.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
snihttphostmatch	<String>	Read-write	snihttphostmatch.
encrypttriggerpktcount	<Integer>	Read-write	encrypttriggerpktcount.
sendclosenotify	<String>	Read-write	sendclosenotify.
poll_time	<Integer>	Read-only	Last Polling Time.
display_name	<String>	Read-only	Display Name.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_sslparameter **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_sslparameter":[{

"denysslreneg":<String_value>,

"insertionencoding":<String_value>,

"defaultprofile":<Boolean_value>,

"dropreqwithnohostheader":<String_value>,
```

```
"strictcachecks":<String_value>,

"pushenctriggertimeout":<Integer_value>,

"quantumsize":<Integer_value>,

"ns_ip_address":<String_value>,

"ssltriggertimeout":<Integer_value>,

"pushflag":<Integer_value>,

"id":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"snihttphostmatch":<String_value>,

"encrypttriggerpktcount":<Integer_value>,

"partition_name":<String_value>,

"sendclosenotify":<String_value>}]}
```

ns_sslprofile

Configuration for Citrix ADC SSL Profile resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
denysslreneg	<String>	Read-write	denysslreneg.
sessionticketkeyrefresh	<String>	Read-write	sessionticketkeyrefresh.
quantumsize	<Integer>	Read-write	quantumsize.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cipherurl	<String>	Read-write	cipherurl.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
ssltriggertimeout	<Integer>	Read-write	ssltriggertimeout.
pushflag	<Integer>	Read-write	pushflag.
allowextendedmastersecret	<String>	Read-write	allowextendedmastersecret.
commonname	<String>	Read-write	commonname.
ssllogprofile	<String>	Read-write	ssllogprofile.
name	<String>	Read-write	Name. Minimum length = 1 Maximum length = 100
dhcount	<Integer>	Read-write	dhcount.
tls13sessionticketsperauthcontext	<Integer>	Read-write	tls13sessionticketsperauthcontext.
pushenctriggertimeout	<Integer>	Read-write	pushenctriggertimeout.
dh	<Boolean>	Read-write	dh.
hsts	<Boolean>	Read-write	hsts.
serverauth	<String>	Read-write	serverauth.
tls13	<Boolean>	Read-write	tls13.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
snienable	<String>	Read-write	snienable.
sslinterception	<String>	Read-write	sslinterception.
cipherredirect	<String>	Read-write	cipherredirect.
strictsigdigestcheck	<Boolean>	Read-write	strictsigdigestcheck.
sslimaxsessperserver	<Integer>	Read-write	sslimaxsessperserver.
clientcert	<String>	Read-write	clientcert.
prevsessionkeylifetime	<Integer>	Read-write	prevsessionkeylifetime.
sessionticketlifetime	<Integer>	Read-write	sessionticketlifetime.
dhfile	<String>	Read-write	dhfile.
encrypttriggerpktcount	<Integer>	Read-write	encrypttriggerpktcount.
ssl2	<Boolean>	Read-write	ssl2.
sslredirect	<String>	Read-write	sslredirect.
insertionencoding	<String>	Read-write	insertionencoding.
ersa	<Boolean>	Read-write	ersa.
strictcachecks	<String>	Read-write	strictcachecks.
sslireneg	<String>	Read-write	sslireneg.
redirectportrewrite	<String>	Read-write	redirectportrewrite.
sessreuse	<String>	Read-write	sessreuse.
ssl3	<Boolean>	Read-write	ssl3.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
sessionticketkeydata	<String>	Read-write	sessionticketkeydata .
sessionticket	<String>	Read-write	sessionticket.
id	<String>	Read-write	Id is system generated key.
cleartextport	<String>	Read-write	cleartextport.
alpnprotocol	<String>	Read-write	alpnprotocol.
clientauthuseboundcachain	<String>	Read-write	clientauthuseboundcachain.
ocspstapling	<String>	Read-write	ocspstapling.
dhekeyexchangewithpsk	<String>	Read-write	dhekeyexchangewithpsk.
ersacount	<Integer>	Read-write	ersacount.
snihttphostmatch	<String>	Read-write	snihttphostmatch.
preload	<String>	Read-write	preload.
sesstimeout	<Integer>	Read-write	sesstimeout.
sendclosenotify	<String>	Read-write	sendclosenotify.
tls11	<Boolean>	Read-write	tls11.
skipclientcertpolicycheck	<String>	Read-write	skipclientcertpolicycheck.
dropreqwithnohostheader	<String>	Read-write	dropreqwithnohostheader.
maxage	<Integer>	Read-write	maxage.
sessionkeylifetime	<Integer>	Read-write	sessionkeylifetime.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
tls12	<Boolean>	Read-write	tls12.
includesubdomains	<Boolean>	Read-write	includesubdomains.
tls10	<Boolean>	Read-write	tls1.
dhkeyexpsizelimit	<String>	Read-write	dhkeyexpsizelimit.
ssliocspcheck	<String>	Read-write	ssliocspcheck.
pushenctrigger	<String>	Read-write	pushenctrigger.
clientauth	<String>	Read-write	clientauth.
zerorttearlydata	<String>	Read-write	zerorttearlydata.
poll_time	<Integer>	Read-only	Last Polling Time.
display_name	<String>	Read-only	Display Name.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_sslprofile **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_sslprofile":[{

"denysslreneg":<String_value>,

"sessionticketkeyrefresh":<String_value>,

"quantumsize":<Integer_value>,

"cipherurl":<String_value>,

"ns_ip_address":<String_value>,

"ssltriggertimeout":<Integer_value>,

"pushflag":<Integer_value>,

"allowextendedmastersecret":<String_value>,

"commonname":<String_value>,

"ssllogprofile":<String_value>,

"name":<String_value>,

"dhcount":<Integer_value>,

"tls13sessionticketsperauthcontext":<Integer_value>,

"pushenctriggertimeout":<Integer_value>,

"dh":<Boolean_value>,

"hsts":<Boolean_value>,

"serverauth":<String_value>,

"tls13":<Boolean_value>,

"snienable":<String_value>,

"sslinterception":<String_value>,
```

```
"cipherredirect":<String_value>,

"strictsigdigestcheck":<Boolean_value>,

"sslimaxsessperserver":<Integer_value>,

"clientcert":<String_value>,

"prevsessionkeylifetime":<Integer_value>,

"sessionticketlifetime":<Integer_value>,

"dhfile":<String_value>,

"encrypttriggerpktcount":<Integer_value>,

"ssl2":<Boolean_value>,

"sslredirect":<String_value>,

"insertionencoding":<String_value>,

"ersa":<Boolean_value>,

"strictcachecks":<String_value>,

"sslireneg":<String_value>,

"redirectportrewrite":<String_value>,

"sessreuse":<String_value>,

"ssl3":<Boolean_value>,

"sessionticketkeydata":<String_value>,

"sessionticket":<String_value>,

"id":<String_value>,

"cleartextport":<String_value>,

"alpnprotocol":<String_value>,
```

```
"clientauthuseboundcachain":<String_value>,

"poll_time":<Integer_value>,

"ocspstapling":<String_value>,

"dhekeyexchangewithpsk":<String_value>,

"ersacount":<Integer_value>,

"snihttphostmatch":<String_value>,

"preload":<String_value>,

"sesstimeout":<Integer_value>,

"sendclosenotify":<String_value>,

"tls11":<Boolean_value>,

"skipclientcertpolicycheck":<String_value>,

"dropreqwithnohostheader":<String_value>,

"maxage":<Integer_value>,

"sessionkeylifetime":<Integer_value>,

"tls12":<Boolean_value>,

"includesubdomains":<Boolean_value>,

"tls10":<Boolean_value>,

"dhkeyexpsizelimit":<String_value>,

"ssliocspcheck":<String_value>,

"pushenctrigger":<String_value>,

"clientauth":<String_value>,

"display_name":<String_value>,
```

```
"zerorttearlydata":<String_value>,

"partition_name":<String_value>}]}
```

ns_sslprofile_cipher_binding

Configuration for Citrix ADC SSL Profile Cipher Binding resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ciphername	<String>	Read-write	ciphername.
cipherpriority	<Integer>	Read-write	cipherpriority.
id	<String>	Read-write	Id is system generated key.
profilename	<String>	Read-write	profilename.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
partition_name	<String>	Read-only	Partition Name.
poll_time	<Integer>	Read-only	Last Polling Time.
display_name	<String>	Read-only	Display Name.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_sslprofile_cipher_binding **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_sslprofile_cipher_binding":[{

"ciphername":<String_value>,

"cipherpriority":<Integer_value>,

"partition_name":<String_value>,

"poll_time":<Integer_value>,

"id":<String_value>,

"profilename":<String_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>}]}
```

ns_sslvserver_cipher_binding

Configuration for Citrix ADC SSL Vserver Cipher Binding resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ciphername	<String>	Read-write	ciphername.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cipherpriority	<Integer>	Read-write	cipherpriority.
sslvserver	<String>	Read-write	sslvserver.
id	<String>	Read-write	Id is system generated key.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
partition_name	<String>	Read-only	Partition Name.
poll_time	<Integer>	Read-only	Last Polling Time.
display_name	<String>	Read-only	Display Name.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_sslvserver_cipher_binding **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_sslvserver_cipher_binding":[{

"ciphername":<String_value>,

"cipherpriority":<Integer_value>,

"partition_name":<String_value>,

"sslvserver":<String_value>,

"poll_time":<Integer_value>,

"id":<String_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>}]}
```

ns_syslog_config

Configuration for Citrix ADC Syslog Config resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
informational	<Boolean>	Read-write	Enable informational.
alert	<Boolean>	Read-write	Enable alert.
none	<Boolean>	Read-write	Enable none.
all	<Boolean>	Read-write	Enable all.
facility	<String>	Read-write	Facility.
state	<String>	Read-write	Syslog Status.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String>	Read-write	Citrix ADC IP Address.
notice	<Boolean>	Read-write	Enable notice.
destination_ip	<String>	Read-write	Syslog Destination IP Address.
critical	<Boolean>	Read-write	Enable critical.
emergency	<Boolean>	Read-write	Enable emergency.
warning	<Boolean>	Read-write	Enable warning.
error	<Boolean>	Read-write	Enable error.
debug	<Boolean>	Read-write	Enable debug.

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_syslog_config **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_syslog_config **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_syslog_config resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_syslog_config **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_syslog_config resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_syslog_config **?count=yes** Use this query-parameter to get the count of ns_syslog_config resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_syslog_config":[{

"informational":<Boolean_value>,

"alert":<Boolean_value>,

"none":<Boolean_value>,

"all":<Boolean_value>,

"facility":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"notice":<Boolean_value>,

"destination_ip":<String_value>,

"critical":<Boolean_value>,

"emergency":<Boolean_value>,

"warning":<Boolean_value>,

"error":<Boolean_value>,

"debug":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_syslog_config/ns_ip_address_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_syslog_config":[{

"informational":<Boolean_value>,

"alert":<Boolean_value>,

"none":<Boolean_value>,

"all":<Boolean_value>,

"facility":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"notice":<Boolean_value>,
```

```
"destination_ip":<String_value>,

"critical":<Boolean_value>,

"emergency":<Boolean_value>,

"warning":<Boolean_value>,

"error":<Boolean_value>,

"debug":<Boolean_value>}]}
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/ns_syslog_config/ns_ip_address_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_syslog_config":{

"ns_ip_address":<String_value>,

"informational":<Boolean_value>,

"all":<Boolean_value>,

"warning":<Boolean_value>,

"facility":<String_value>,

"destination_ip":<String_value>,
```

```
"debug":<Boolean_value>,

"none":<Boolean_value>,

"notice":<Boolean_value>,

"error":<Boolean_value>,

"alert":<Boolean_value>,

"state":<String_value>,

"critical":<Boolean_value>,

"emergency":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_syslog_config":[{

"informational":<Boolean_value>,

"alert":<Boolean_value>,

"none":<Boolean_value>,

"all":<Boolean_value>,

"facility":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"notice":<Boolean_value>,
```

```
"destination_ip":<String_value>,

"critical":<Boolean_value>,

"emergency":<Boolean_value>,

"warning":<Boolean_value>,

"error":<Boolean_value>,

"debug":<Boolean_value>}]}
```

ns_syslog_task

Configuration for Citrix ADC Syslog Configure Task resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
scheduleTimesEpoch	<String>	Read-write	Schedule time epoch (string representation of 11 digit numbers)..
device_family	<String>	Read-write	Family of Devices for which config template is defined.. Minimum length = 1 Maximum length = 64
device_groups	<String[]>	Read-write	Device Group Array on which for which template is applied.
scheduleTimes	<String>	Read-write	Comma separated times of the day(HH:MM) on which Configuration

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Template is scheduled.
variables	<variable_values[]>	Read-write	Values of Variables used in commands of the configuration template.
on_error	<String>	Read-write	Behaviour on encountering error while applying configuration template on a device: CONTINUE| EXIT. Minimum length = 1 Maximum length = 16
port	<Integer>	Read-write	Syslog server port. Maximum value =
execute_sequentially	<Boolean>	Read-write	True if configuration template is to be applied to devices sequentially.
tz_offset	<Integer>	Read-write	Time zone offset..
scheduleOptions	<String>	Read-write	Comma separated Options for scheduling Configuration Template.
devices	<String[]>	Read-write	Device IP Address Array on which template is applied.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
scheduleType	<String>	Read-write	Schedule Type of Configuration Template that is scheduled.
ip_address	<String>	Read-write	Syslog server IP address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns_syslog_task?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_syslog_task": {

<b>"port":<Integer_value></b>,

<b>"ip_address":<String_value></b>,

"scheduleTimes":<String_value>,

"execute_sequentially":<Boolean_value>,

"scheduleTimesEpoch":<String_value>,

"device_family":<String_value>,
```

```
"on_error":<String_value>,

"devices":<String_value[]>,

"scheduleOptions":<String_value>,

"device_groups":<String_value[]>,

"scheduleType":<String_value>,

"tz_offset":<Integer_value>,

"variables":[{

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"values_enum_db":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"description":<String_value>,

"type":<String_value>,

"valuex":<String_value>,

"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"device":<String_value>,
```

```
"device_group":<String_value>,

"valuex":<String_value>,

"parent_id":<String_value>}],

"display_name":<String_value>}]}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_syslog_task":[{

"scheduleTimesEpoch":<String_value>,

"device_family":<String_value>,

"device_groups":<String_value>,

"scheduleTimes":<String_value>,

"variables":[{

"device_values_mapping":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,
```

```
"device_values":[{

"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"valuex":<String_value>,

"device_group":<String_value>,

"device":<String_value>,

"parent_id":<String_value>}],

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>}],

"on_error":<String_value>,

"port":<Integer_value>,

"execute_sequentially":<Boolean_value>,

"tz_offset":<Integer_value>,

"scheduleOptions":<String_value>,

"devices":<String_value>,

"scheduleType":<String_value>,

"ip_address":<String_value>}]}
```

```
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/ns_syslog_task/ **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_syslog_task **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_syslog_task **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_syslog_task resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_syslog_task **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_syslog_task resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_syslog_task **?count=yes** Use this query-parameter to get the count of ns_syslog_task resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_syslog_task":[{

"scheduleTimesEpoch":<String_value>,

"device_family":<String_value>,

"device_groups":<String_value>,

"scheduleTimes":<String_value>,

"variables":[{

"device_values_mapping":<String_value>,

"parent_name":<String_value>,

"value":<String_value>,

"name":<String_value>,

"valuex":<String_value>,

"display_name":<String_value>,

"description":<String_value>,

"device_values":[{
```

```
"parent_name":<String_value>,

"value":<String_value>,

"id":<String_value>,

"valuex":<String_value>,

"device_group":<String_value>,

"device":<String_value>,

"parent_id":<String_value>}],

"parent_id":<String_value>,

"default_value":<String_value>,

"id":<String_value>,

"type":<String_value>,

"values_enum_db":<String_value>}],

"on_error":<String_value>,

"port":<Integer_value>,

"execute_sequentially":<Boolean_value>,

"tz_offset":<Integer_value>,

"scheduleOptions":<String_value>,

"devices":<String_value>,

"scheduleType":<String_value>,

"ip_address":<String_value>}]}
```

ns_upgrade

Configuration for Upgrade Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	image_name. Maximum length = 128
cleanup_unsupported_policy	<Boolean>	Read-write	True, if user selected to delete unsupported policies after upgrade..
slack_profile	<String>	Read-write	Slack profile.
id	<String>	Read-write	Id is maintenance job id.
do_cleanup	<Boolean>	Read-write	True, if user wants to clean the image files once upgrade is completed successfully.
scheduleTimesEpoch	<String>	Read-write	Schedule time epoch (string representation of 11 digit numbers)..
doc_file	<String>	Read-write	Documentation File Name. Minimum length = 1 Maximum length = 128
name	<String>	Read-write	Name of upgrade. Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
do_backup	<Boolean>	Read-write	True, if user wants to take backup before ns upgrade.
ns_ip_address_arr	<String[]>	Read-write	List of Citrix ADC IP Address.
is_issu_enabled	<Boolean>	Read-write	True, if ISSU is enabled.
mail_profiles	<String>	Read-write	Comma separated list of Mail profiles.
scheduleId	<String>	Read-write	scheduleId is used to refer maintenaince schedule task.
post_upgrade_script	<String>	Read-write	Name of post upgrade script file. Maximum length = 128
is_post_upgrade_pre_failover_script_selected	<Boolean>	Read-write	True, if post upgrade pre failover custom script is selected.
saveconfig_enabled	<Boolean>	Read-write	True, if save config before upgrade.
pre_upgrade_script	<String>	Read-write	Name of pre upgrade script file. Maximum length = 128
adc_cloud_image_version	<String>	Read-write	ADC Cloud Image Version to which AutoScaleGroup(s) is/are to be upgraded. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
issu_migration_timeout	<Integer>	Read-write	ISSU migration timeout value in minutes. Minimum value = 10 Maximum value =
is_post_upgrade_script_selected	<Boolean>	Read-write	True, if post upgrade custom script is selected.
device_groups	<String[]>	Read-write	Device Group Array on which for which job is run.
is_remote_download	<Boolean>	Read-write	True if upgrade is remote download instead of direct upload from ADM.
post_upgrade_pre_failover_script	<String>	Read-write	Name of post upgrade pre failover script file. Maximum length = 128
ha_two_phase_upgrade	<Boolean>	Read-write	True, if user wants to upgarde ha in two phases or steps.
second_failover_enabled	<Boolean>	Read-write	True, if second failover for HA upgrade is enabled.
is_pre_post_upgrade_script_same	<Boolean>	Read-write	True, if pre and post upgrade post failover custom scripts are same.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ha_node2_devices	<String[]>	Read-write	List of HA Citrix ADC IP Address upgrade in second phase.
adc_cloud_image_id	<String>	Read-write	ADC Cloud Image ID to which AutoScaleGroup(s) is/are to be upgraded. Minimum length = 1 Maximum length = 128
is_pre_upgrade_script_selected	<Boolean>	Read-write	True, if pre upgrade custom script is selected.
scheduleTimesEpoch2	<String>	Read-write	Schedule time epoch for ha primary node (string representation of 11 digit numbers)..
is_pre_post_upgrade_pre_failover_script_same	<Boolean>	Read-write	True, if pre and post upgrade pre failover custom scripts are same.
act_id	<String>	Read-only	Activity Id.
tasklog_id	<String>	Read-only	Task Log Id of the Config Job.

Operations

(click to see Properties)

- GET (ALL)
- UPGRADE
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_upgrade **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_upgrade **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_upgrade resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_upgrade **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_upgrade resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_upgrade **?count=yes** Use this query-parameter to get the count of ns_upgrade resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_upgrade":[{

"mode":<Integer_value>,

"image_name":<String_value>,

"cleanup_unsupported_policy":<Boolean_value>,

"slack_profile":<String_value>,

"act_id":<String_value>,

"id":<String_value>,
```

```
"do_cleanup":<Boolean_value>,

"scheduleTimesEpoch":<String_value>,

"tasklog_id":<String_value>,

"doc_file":<String_value>,

"name":<String_value>,

"do_backup":<Boolean_value>,

"ns_ip_address_arr":<String_value>,

"is_issu_enabled":<Boolean_value>,

"mail_profiles":<String_value>,

"scheduleId":<String_value>,

"post_upgrade_script":<String_value>,

"is_post_upgrade_pre_failover_script_selected":<Boolean_value>,

"saveconfig_enabled":<Boolean_value>,

"pre_upgrade_script":<String_value>,

"adc_cloud_image_version":<String_value>,

"issu_migration_timeout":<Integer_value>,

"is_post_upgrade_script_selected":<Boolean_value>,

"device_groups":<String_value>,

"is_remote_download":<Boolean_value>,

"post_upgrade_pre_failover_script":<String_value>,

"ha_two_phase_upgrade":<Boolean_value>,

"second_failover_enabled":<Boolean_value>,
```

```
"is_pre_post_upgrade_script_same":<Boolean_value>,

"ha_node2_devices":<String_value>,

"adc_cloud_image_id":<String_value>,

"is_pre_upgrade_script_selected":<Boolean_value>,

"scheduleTimesEpoch2":<String_value>,

"is_pre_post_upgrade_pre_failover_script_same":<Boolean_value>}]}
```

UPGRADE

URL: https://<MGMT-IP>/nitro/v2/config/ns_upgrade?action=upgrade;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_upgrade": {

<b>"image_name":<String_value></b>,

"name":<String_value>,

"saveconfig_enabled":<Boolean_value>,

"is_post_upgrade_script_selected":<Boolean_value>,

"post_upgrade_pre_failover_script":<String_value>,

"ha_two_phase_upgrade":<Boolean_value>,
```

```
"second_failover_enabled":<Boolean_value>,

"is_pre_post_upgrade_script_same":<Boolean_value>,

"do_cleanup":<Boolean_value>,

"id":<String_value>,

"scheduleTimesEpoch":<String_value>,

"ns_ip_address_arr":<String_value[]>,

"is_issu_enabled":<Boolean_value>,

"mail_profiles":<String_value>,

"pre_upgrade_script":<String_value>,

"issu_migration_timeout":<Integer_value>,

"device_groups":<String_value[]>,

"adc_cloud_image_id":<String_value>,

"ha_node2_devices":<String_value[]>,

"do_backup":<Boolean_value>,

"doc_file":<String_value>,

"is_post_upgrade_pre_failover_script_selected":<Boolean_value>,

"adc_cloud_image_version":<String_value>,

"is_remote_download":<Boolean_value>,

"is_pre_upgrade_script_selected":<Boolean_value>,

"cleanup_unsupported_policy":<Boolean_value>,

"slack_profile":<String_value>,

"scheduleId":<String_value>,
```

```
"post_upgrade_script":<String_value>,

"is_pre_post_upgrade_pre_failover_script_same":<Boolean_value>,

"scheduleTimesEpoch2":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_upgrade":[{

"mode":<Integer_value>,

"image_name":<String_value>,

"cleanup_unsupported_policy":<Boolean_value>,

"slack_profile":<String_value>,

"act_id":<String_value>,

"id":<String_value>,

"do_cleanup":<Boolean_value>,

"scheduleTimesEpoch":<String_value>,

"tasklog_id":<String_value>,

"doc_file":<String_value>,

"name":<String_value>,

"do_backup":<Boolean_value>,

"ns_ip_address_arr":<String_value>,
```

```
"is_issu_enabled":<Boolean_value>,

"mail_profiles":<String_value>,

"scheduleId":<String_value>,

"post_upgrade_script":<String_value>,

"is_post_upgrade_pre_failover_script_selected":<Boolean_value>,

"saveconfig_enabled":<Boolean_value>,

"pre_upgrade_script":<String_value>,

"adc_cloud_image_version":<String_value>,

"issu_migration_timeout":<Integer_value>,

"is_post_upgrade_script_selected":<Boolean_value>,

"device_groups":<String_value>,

"is_remote_download":<Boolean_value>,

"post_upgrade_pre_failover_script":<String_value>,

"ha_two_phase_upgrade":<Boolean_value>,

"second_failover_enabled":<Boolean_value>,

"is_pre_post_upgrade_script_same":<Boolean_value>,

"ha_node2_devices":<String_value>,

"adc_cloud_image_id":<String_value>,

"is_pre_upgrade_script_selected":<Boolean_value>,

"scheduleTimesEpoch2":<String_value>,

"is_pre_post_upgrade_pre_failover_script_same":<Boolean_value>}]}
```

MODIFY

URL: https://<MGMT-IP>/nitro/v2/config/ns_upgrade/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_upgrade":{

<b>"image_name":<String_value></b>,

"name":<String_value>,

"saveconfig_enabled":<Boolean_value>,

"is_post_upgrade_script_selected":<Boolean_value>,

"post_upgrade_pre_failover_script":<String_value>,

"ha_two_phase_upgrade":<Boolean_value>,

"second_failover_enabled":<Boolean_value>,

"is_pre_post_upgrade_script_same":<Boolean_value>,

"do_cleanup":<Boolean_value>,

"id":<String_value>,

"scheduleTimesEpoch":<String_value>,

"ns_ip_address_arr":<String_value[]>,

"is_issu_enabled":<Boolean_value>,
```

```
"mail_profiles":<String_value>,

"pre_upgrade_script":<String_value>,

"issu_migration_timeout":<Integer_value>,

"device_groups":<String_value[]>,

"adc_cloud_image_id":<String_value>,

"ha_node2_devices":<String_value[]>,

"do_backup":<Boolean_value>,

"doc_file":<String_value>,

"is_post_upgrade_pre_failover_script_selected":<Boolean_value>,

"adc_cloud_image_version":<String_value>,

"is_remote_download":<Boolean_value>,

"is_pre_upgrade_script_selected":<Boolean_value>,

"cleanup_unsupported_policy":<Boolean_value>,

"slack_profile":<String_value>,

"scheduleId":<String_value>,

"post_upgrade_script":<String_value>,

"is_pre_post_upgrade_pre_failover_script_same":<Boolean_value>,

"scheduleTimesEpoch2":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_upgrade":[{

"mode":<Integer_value>,

"image_name":<String_value>,

"cleanup_unsupported_policy":<Boolean_value>,

"slack_profile":<String_value>,

"act_id":<String_value>,

"id":<String_value>,

"do_cleanup":<Boolean_value>,

"scheduleTimesEpoch":<String_value>,

"tasklog_id":<String_value>,

"doc_file":<String_value>,

"name":<String_value>,

"do_backup":<Boolean_value>,

"ns_ip_address_arr":<String_value>,

"is_issu_enabled":<Boolean_value>,

"mail_profiles":<String_value>,

"scheduleId":<String_value>,

"post_upgrade_script":<String_value>,

"is_post_upgrade_pre_failover_script_selected":<Boolean_value>,

"saveconfig_enabled":<Boolean_value>,

"pre_upgrade_script":<String_value>,
```

```
"adc_cloud_image_version":<String_value>,

"issu_migration_timeout":<Integer_value>,

"is_post_upgrade_script_selected":<Boolean_value>,

"device_groups":<String_value>,

"is_remote_download":<Boolean_value>,

"post_upgrade_pre_failover_script":<String_value>,

"ha_two_phase_upgrade":<Boolean_value>,

"second_failover_enabled":<Boolean_value>,

"is_pre_post_upgrade_script_same":<Boolean_value>,

"ha_node2_devices":<String_value>,

"adc_cloud_image_id":<String_value>,

"is_pre_upgrade_script_selected":<Boolean_value>,

"scheduleTimesEpoch2":<String_value>,

"is_pre_post_upgrade_pre_failover_script_same":<Boolean_value>}]}
```

ns_version

Configuration for Citrix ADC Version Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
version	<String>	Read-write	The contact person's E-mail address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 78
ip_address	<String>	Read-write	Citrix ADC IP Address for callhome. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_version **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_version **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_version resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_version **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_version resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_version **?count=yes** Use this query-parameter to get the count of ns_version resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_version":[{

"version":<String_value>,

"ip_address":<String_value>}]}
```

ns_vlan

Configuration for Citrix ADC Vlan resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
bound_ifaces	<String>	Read-write	Bounded Interfaces.
ns_vlan_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
tagged_ifaces	<String>	Read-write	Tagged Interfaces.
if_ipv6_routing	<String>	Read-write	VLAN for Network 1/8 on VM Instance.
ns_vlan_id	<Integer>	Read-write	VLAN Id. Maximum value =

Operations

(click to see Properties)

- DELETE
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/ns_vlan/ **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_vlan **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_vlan **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_vlan resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_vlan **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_vlan resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_vlan **?count=yes** Use this query-parameter to get the count of ns_vlan resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vlan":[{

"bound_ifaces":<String_value>,

"ns_vlan_ip_address":<String_value>,

"tagged_ifaces":<String_value>,

"if_ipv6_routing":<String_value>,

"ns_vlan_id":<Integer_value>}]}
```

ns_vm_image

Configuration for Citrix ADC VM Image File For All Supported Hypervisor Types resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- GET (ALL)
- GET

- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_vm_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_vm_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of

ns_vm_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_vm_image **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_vm_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_vm_image **?count=yes** Use this query-parameter to get the count of ns_vm_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vm_image":[{

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_vm_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vm_image":[{

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

DOWNLOAD

URL: https://<MGMT-IP>/nitro/v2/download/ns_vm_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload: Binary Stream

ns_vmacs

Configuration for vmacs for shared vlans resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vmacs	<String>	Read-write	Comma separated VMac list. Minimum length = 1
additional_mac_free_count	<Integer>	Read-write	Place holder for to view additional mac count.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
increment_by	<Integer>	Read-write	increment by. Minimum value = 1 Maximum value =
generation_method	<String>	Read-write	(MANUAL, BASE or RANDOM).

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 4 Maximum length = 16
vmac_count	<Integer>	Read-write	increment by. Minimum value = 1
sync_operation	<Boolean>	Read-write	sync operation.
base_vmac	<String>	Read-write	Base VMac Address. Minimum length = 1 Maximum length = 32
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- ADD
- REMOVE
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns_vmacs?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_vmacs": {

<b>"ns_ip_address":<String_value></b>,

<b>"generation_method":<String_value></b>,

"increment_by":<Integer_value>,
```

```
"base_vmac":<String_value>,

"sync_operation":<Boolean_value>,

"vmacs":<String_value>,

"vmac_count":<Integer_value>,

"additional_mac_free_count":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vmacs":[{

"vmacs":<String_value>,

"additional_mac_free_count":<Integer_value>,

"ns_ip_address":<String_value>,

"increment_by":<Integer_value>,

"generation_method":<String_value>,

"vmac_count":<Integer_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>,

"base_vmac":<String_value>}]}
```

REMOVE

URL: https://<MGMT-IP>/nitro/v2/config/ns_vmacs?action=remove;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_vmacs": {

"ns_ip_address":<String_value>,

"increment_by":<Integer_value>,

"base_vmac":<String_value>,

"sync_operation":<Boolean_value>,

"generation_method":<String_value>,

"vmacs":<String_value>,

"vmac_count":<Integer_value>,

"additional_mac_free_count":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vmacs":[{

"vmacs":<String_value>,

"additional_mac_free_count":<Integer_value>,

"ns_ip_address":<String_value>,

"increment_by":<Integer_value>,

"generation_method":<String_value>,

"vmac_count":<Integer_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>,

"base_vmac":<String_value>}]}
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_vmacs **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_vmacs **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_vmacs resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_vmacs **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_vmacs resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_vmacs **?count=yes** Use this query-parameter to get the count of ns_vmacs resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vmacs":[{

"vmacs":<String_value>,

"additional_mac_free_count":<Integer_value>,

"ns_ip_address":<String_value>,

"increment_by":<Integer_value>,

"generation_method":<String_value>,

"vmac_count":<Integer_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>,

"base_vmac":<String_value>}]}
```

ns_vmacs_free_count

Configuration for Citrix ADC VMACs free count resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
free_count	<Integer>	Read-write	Citrix ADC VMAC free count.
ip_address	<String>	Read-write	Citrix ADC/VM IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_vmacs_free_count **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_vmacs_free_count **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_vmacs_free_count resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_vmacs_free_count **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_vmacs_free_count resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_vmacs_free_count **?count=yes** Use this query-parameter to get the count of ns_vmacs_free_count resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vmacs_free_count":[{

"free_count":<Integer_value>,

"ip_address":<String_value>}]}
```

ns_vserver_appflow_config

Configuration for Virtual server appflow config on Citrix ADC resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
tenant_id	<String>	Read-write	TenantID.
agent_list	<String[]>	Read-write	Agent List, on which traffic will flow.
ip_address	<String>	Read-write	Virtual server IP Address. Minimum length = 1 Maximum length = 64
icalog	<String>	Read-write	ICA log.
servicetype	<String>	Read-write	servicetype.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
export_option	<String[]>	Read-write	Export Options: TCP, ICA.
appflowlog	<String>	Read-write	Appflow log.
appflow_policy_rule	<String>	Read-write	Appflow policy rule.
name	<String>	Read-write	Virtual server name. Maximum length = 255
es4nslog	<String>	Read-write	ESNS enable.
export_list	<String[]>	Read-write	Export List: WebInsight, SecurityInsight.
transport_mode	<String>	Read-write	Transport Mode [Logstream/ Appflow].
type	<String>	Read-write	Tells whether virtual server type. Minimum length = 2 Maximum length = 64
state	<String>	Read-only	Tells whether virtual server up/down.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/ns_vserver_appflow_config?onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_vserver_appflow_config": {

"ns_ip_address":<String_value>,

"servicetype":<String_value>,

"name":<String_value>,

"export_list":<String_value[]>,

"type":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"tenant_id":<String_value>,

"agent_list":<String_value[]>,

"export_option":<String_value[]>,

"ip_address":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vserver_appflow_config":[{

"state":<String_value>,

"ns_ip_address":<String_value>,

"tenant_id":<String_value>,

"agent_list":<String_value>,

"ip_address":<String_value>,

"icalog":<String_value>,

"servicetype":<String_value>,

"export_option":<String_value>,

"primary_appflowlog":<String_value>,

"appflowlog":<String_value>,

"appflow_policy_rule":<String_value>,

"name":<String_value>,

"es4nslog":<String_value>,

"export_list":<String_value>,

"is_vsever_to_query":<Boolean_value>,

"transport_mode":<String_value>,

"type":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/ns_vserver_appflow_config/ **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_vserver_appflow_config **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_vserver_appflow_config **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_vserver_appflow_config resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_vserver_appflow_config **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_vserver_appflow_config resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_vserver_appflow_config **?count=yes** Use this query-parameter to get the count of ns_vserver_appflow_config resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vserver_appflow_config":[{

"state":<String_value>,

"ns_ip_address":<String_value>,

"tenant_id":<String_value>,

"agent_list":<String_value>,

"ip_address":<String_value>,

"icalog":<String_value>,

"servicetype":<String_value>,

"export_option":<String_value>,

"primary_appflowlog":<String_value>,

"appflowlog":<String_value>,

"appflow_policy_rule":<String_value>,

"name":<String_value>,

"es4nslog":<String_value>,

"export_list":<String_value>,

"is_vsever_to_query":<Boolean_value>,
```

```
"transport_mode":<String_value>,

"type":<String_value>}]}
```

MODIFY

URL: https://<MGMT-IP>/nitro/v2/config/ns_vserver_appflow_config/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_vserver_appflow_config":{

"ns_ip_address":<String_value>,

"servicetype":<String_value>,

"name":<String_value>,

"export_list":<String_value[]>,

"type":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"tenant_id":<String_value>,

"agent_list":<String_value[]>,
```

```
"export_option":<String_value[]>,

"ip_address":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vserver_appflow_config":[{

"state":<String_value>,

"ns_ip_address":<String_value>,

"tenant_id":<String_value>,

"agent_list":<String_value>,

"ip_address":<String_value>,

"icalog":<String_value>,

"servicetype":<String_value>,

"export_option":<String_value>,

"primary_appflowlog":<String_value>,

"appflowlog":<String_value>,

"appflow_policy_rule":<String_value>,

"name":<String_value>,

"es4nslog":<String_value>,
```

```
"export_list":<String_value>,

"is_vsever_to_query":<Boolean_value>,

"transport_mode":<String_value>,

"type":<String_value>}]}
```

Citrix-SD-WAN

- br
- br_adduser
- br_appflowconfig
- br_configfile
- br_configurealert
- br_configurebandwidth5x
- br_configurebandwidth6x
- br_configuresyslog
- br_currentconfig
- br_device_profile
- br_disable
- br_enable
- br_image
- br_locallicense
- br_remotelicense
- br_replicateconfig
- br_restart
- br_restoreconfig
- br_snmpmanager
- br_upgrade

br

Configuration for Citrix SD-WAN Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
throughput_limit	<Double>	Read-write	Throughput Limit in Mbps set for VM Instance.
vlan_id_0_1	<Integer>	Read-write	VLAN id for the management interface 0/1. Maximum value =
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
plugin_ip_address	<String>	Read-write	Plugin IP Address. Minimum length = 1 Maximum length = 64
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1 Maximum length = 64
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.
apa_netmask	<String>	Read-write	aPA Netmask. Minimum length = 1 Maximum length = 64
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.
host_ip_address	<String>	Read-write	Host IPAddress where VM is provisioned. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key for all the managed devices.
ipv6_address	<String>	Read-write	IPv6 Address.
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
number_of_acu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
device_family	<String>	Read-write	Device Family. Minimum length = 1 Maximum length = 64
type	<String>	Read-write	Type of device, (Xen | NS).

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	Assign throughput in Mbps to VM Instance.
template_name	<String>	Read-write	Template Name, This parameter is used while provisioning VM Instance with template, template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64
number_of_scu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
domain_name	<String>	Read-write	Domain name of VM Device. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	Image Name, This parameter is used while provisioning VM Instance with XVA image, template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256
vm_memory_total	<Double>	Read-write	Total Memory of VM Instance in MB. 2048MB, 5120MB.
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
description	<String>	Read-write	Description of managed device. Minimum length = 1 Maximum length = 512

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_list	<String>	Read-write	Specify the list of Repeater IPs for which this device will act as a broker.. Minimum length = 1 Maximum length = 1024
fips_partition_name	<String>	Read-write	FIPS Partition Name. Minimum length = 1 Maximum length = 128
burst_priority	<Integer>	Read-write	Burst Priority of the VM Instance between 1 and 4.
is_managed	<Boolean>	Read-write	Is Managed.
apa_gateway	<String>	Read-write	aPA Gateway. Minimum length = 1 Maximum length = 64
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
network_interfaces	<network_interface[]>	Read-write	Network Interfaces.
last_updated_time	<Double>	Read-write	Last Updated Time.
license_edition	<String>	Read-write	Edition of instance.
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..
la_mgmt	<Boolean>	Read-write	Bond consisting of management ports on VM Instance.
vlan_id_0_2	<Integer>	Read-write	VLAN id for the management interface 0/2. Maximum value =
number_of_ssl_cores	<Integer>	Read-write	Assign number of ssl virtual functions to VM Instance.
max_burst_throughput	<Double>	Read-write	Maximum burst throughput in Mbps of VM Instance.
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal network (not required to be set for SDWAN).

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
node_id	<String>	Read-write	Node identification of a device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1 Maximum length = 64
throughput_allocation_mode	<Integer>	Read-write	Throughput Allocation Mode: 0-Fixed, 1-Burst-able.
number_of_cores	<Integer>	Read-write	Number of cores that are assigned to VM Instance.
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be A-B, and for Cluster it will be CLIP. Minimum length = 1 Maximum length = 128
std_bw_total	<Integer>	Read-write	Standard Bandwidth.
apa_ip_address	<String>	Read-write	aPA IP Address. Minimum length = 1 Maximum length = 64
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.
vlan_1_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/2 on VM Instance. Maximum value =
vrid_list_ipv4_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV4 VMAC Generation.
vlan_1_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/7 on VM Instance. Maximum value =
vrid_list_ipv6_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV6 VMAC Generation.
vrid_list_ipv4_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 1/4 for IPV4 VMAC Generation.
vrid_list_ipv6_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV6 VMAC Generation.
if_10_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/4 on VM Instance.
vlan_10_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/5 on VM Instance. Maximum value =
vrid_list_ipv6_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV6 VMAC Generation.
vrid_list_ipv6_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV6 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/5 on VM Instance.
receiveuntagged_1_ 4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/4 on VM Instance.
vrid_list_ipv6_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV6 VMAC Generation.
vrid_list_ipv6_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV6 VMAC Generation.
if_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/3 on VM Instance.
vrid_list_ipv6_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			IPV6 VMAC Generation.
receiveuntagged_10_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/4 on VM Instance.
vrid_list_ipv6_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV6 VMAC Generation.
vlan_10_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/6 on VM Instance. Maximum value =
if_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/5 on VM Instance.
is_new_crypto	<Boolean>	Read-write	True if number_of_acu/ number_of_scu are used, false if number_of_ssl_core s is used.
vrid_list_ipv6_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVRID List for Interface 1/6 for IPV6 VMAC Generation.
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
if_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/7 on VM Instance.
receiveuntagged_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/8 on VM Instance.
vrid_list_ipv6_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV6 VMAC Generation.
receiveuntagged_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/6 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_1_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/4 on VM Instance. Maximum value =
if_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/2 on VM Instance.
receiveuntagged_1_ 5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/5 on VM Instance.
receiveuntagged_10 _6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/6 on VM Instance.
receiveuntagged_10 _1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/1 on VM Instance.
vrid_list_ipv4_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVRID List for Interface 1/6 for IPV4 VMAC Generation.
vrid_list_ipv6_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV6 VMAC Generation.
vrid_list_ipv6_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV6 VMAC Generation.
vrid_list_ipv4_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV4 VMAC Generation.
receiveuntagged_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/2 on VM Instance.
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv4_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV4 VMAC Generation.
receiveuntagged_1_ 7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/7 on VM Instance.
vlan_10_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/4 on VM Instance. Maximum value =
if_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/7 on VM Instance.
receiveuntagged_1_ 1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/1 on VM Instance.
vrid_list_ipv6_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 1/7 for IPV6 VMAC Generation.
if_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/1 on VM Instance.
if_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/8 on VM Instance.
l2_enabled	<Boolean>	Read-write	L2mode status of VM Instance.
vlan_10_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/7 on VM Instance. Maximum value =
vlan_10_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/1 on VM Instance. Maximum value =
receiveuntagged_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/2 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
reboot_vm_on_cpu_change	<Boolean>	Read-write	Reboot VMs on CPU change during resource allocation.
vlan_1_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/6 on VM Instance. Maximum value =
vrid_list_ipv4_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV4 VMAC Generation.
receiveuntagged_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/7 on VM Instance.
vrid_list_ipv4_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV4 VMAC Generation.
vlan_1_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/1 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum value =
vrid_list_ipv6_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV6 VMAC Generation.
vlan_1_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/8 on VM Instance. Maximum value =
vrid_list_ipv4_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV4 VMAC Generation.
vlan_10_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/2 on VM Instance. Maximum value =
receiveuntagged_10 _5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/5 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv4_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV4 VMAC Generation.
vrid_list_ipv4_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV4 VMAC Generation.
receiveuntagged_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/8 on VM Instance.
vrid_list_ipv4_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV4 VMAC Generation.
vlan_1_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/3 on VM Instance. Maximum value =
receiveuntagged_10_3	<Boolean>	Read-write	This property is deprecated by

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			network_interfaces;lt br;gtReceive Untagged Packets on 10/3 on VM Instance.
if_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/6 on VM Instance.
if_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/6 on VM Instance.
vlan_10_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/3 on VM Instance. Maximum value =
vlan_1_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/5 on VM Instance. Maximum value =
if_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/3 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv4_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV4 VMAC Generation.
if_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/4 on VM Instance.
vrid_list_ipv4_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV4 VMAC Generation.
if_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/1 on VM Instance.
vrid_list_ipv4_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV4 VMAC Generation.
vlan_10_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 10/8 on VM Instance. Maximum value =
receiveuntagged_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/3 on VM Instance.
crypto_change_requires_reboot	<Boolean>	Read-write	True if the current changes made by user requires a reboot of the VM else false.
vrid_list_ipv4_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for IPV4 VMAC Generation.
if_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/8 on VM Instance.
vrid_list_ipv4_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for IPV6 VMAC Generation.
enable_quick_install	<Boolean>	Read-write	Enables Citrix SD-WAN quick install page.
if_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/2 on VM Instance.
vrid_list_ipv6_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV6 VMAC Generation.
bandwidth_mode	<String>	Read-only	Boost status : Bandwidth Mode of Citrix SD-WAN Instance.
instance_state	<String>	Read-only	State of device, UP only if device accessible.
disk_space	<Double>	Read-only	Show Disk Space (GB) available to VM Instance.
is_active	<Boolean>	Read-only	Is Licensed.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cpu_core_pe	<String>	Read-only	Packet Engine cores assigned to VM Instance.
qosstatus	<String>	Read-only	QoS Status.
wan_in	<Double>	Read-only	WAN In of Citrix SD-WAN Instance in Mbps.
preferred_cpu_socket	<String>	Read-only	Preferred CPU Socket.
device_uuid	<String>	Read-only	Device UUID.
version	<String>	Read-only	Device Version.
vm_memory_used	<Double>	Read-only	Memory Used (MB) by VM Instance.
sysid	<String>	Read-only	System ID.
vm_state	<String>	Read-only	State of Virtual Machine (Running | Halted).
uuid	<String>	Read-only	UUID of VM Instance generated by Hypervisor.
connected_plugins	<String>	Read-only	connected plugins.
cpu_core_mgmt	<String>	Read-only	Management CPU cores assigned to VM Instance.
ssl_virtual_functions	<String>	Read-only	SSL Virtual Functions assigned to VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ha_peer_state	<String>	Read-only	Peer State of Citrix SD-WAN Instance in HA mode.
lan_out_prev	<Double>	Read-only	Lan In of Citrix SD-WAN Instance in Mbps.
wan_in_prev	<Double>	Read-only	WAN In of Citrix SD-WAN Instance in Mbps.
bypass	<String>	Read-only	bypass value of Citrix SD-WAN Instance.
status	<String>	Read-only	Status of managed device.
max_plugins	<String>	Read-only	Maximum plugins.
systemname	<String>	Read-only	Device System Name.
mac_eth0	<String>	Read-only	MAC Address of eth0 on VM Instance.
wan_out	<Double>	Read-only	WAN Out of Citrix SD-WAN Instance in Mbps.
vm_tx	<Double>	Read-only	Out Throughput of VM Instance in Mbps.
prev_poll_time	<Double>	Read-only	Previous time.
unacl_connections	<String>	Read-only	non-accelerated connections.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
license	<String>	Read-only	Citrix SD-WAN License.
acl_connections	<String>	Read-only	Accelerated Connections.
vm_memory_free	<Double>	Read-only	Free Memory (MB) available in VM Instance.
br_memory_usage	<Double>	Read-only	Memory Usage (%) of Citrix SD-WAN Instance.
vm_memory_usage	<Double>	Read-only	Memory Usage (%) of VM Instance.
vm_cpu_usage	<Double>	Read-only	CPU Usage (%) of VM Instance.
sdxtools_http_port	<Integer>	Read-only	Http Port number to communicate to SDXTools.
wan_out_prev	<Double>	Read-only	WAN In of Citrix SD-WAN Instance in Mbps.
reason	<String>	Read-only	Reason of failure for this managed device.
virtual_functions	<String>	Read-only	Virtual Functions assigned to VM Instance depending on SDX platform.
upsince	<String>	Read-only	Upsince of managed device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
domid	<Double>	Read-only	DOM Id assigned to VM Instance viewed in hypervisor.
lan_out	<Double>	Read-only	LAN Out of Citrix SD-WAN Instance in Mbps.
active_connections	<Double>	Read-only	Total Active Connections on Citrix SD-WAN Instance.
tenant_id	<String>	Read-only	Tenant ID.
do_config	<Boolean>	Read-only	Do default config for managed device.
host_id	<String>	Read-only	Host ID.
has_sdxtools	<Boolean>	Read-only	True if SDX Tools are installed on this VM.
system_load	<Double>	Read-only	System Load (%) on Citrix SD-WAN Instance.
number_of_cpu	<Integer>	Read-only	Number of CPU that is assigned to VM Instance.
ha_state	<String>	Read-only	State of Citrix SD-WAN Instance in HA mode.
operation_status	<String>	Read-only	Operation status of Citrix SD-WAN Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_swg	<Boolean>	Read-only	Boolean to indicate whether a VM is SWG.
data_reduction	<Double>	Read-only	Data Reduction (%).
port_model	<Integer>	Read-only	Port Model for BR instance.
uptime	<String>	Read-only	Uptime of device.
ha_peer_ip_address	<String>	Read-only	Peer IP Address of Citrix SD-WAN Instance.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.
havmip	<String>	Read-only	HA VM IPAddress.
lan_in_prev	<Double>	Read-only	Lan In of Citrix SD-WAN Instance in Mbps.
assigned_cpu_socket	<String>	Read-only	Assigned CPU Socket 0 or 1.
lan_in	<Double>	Read-only	Lan In of Citrix SD-WAN Instance in Mbps.
ha_peer_id	<Double>	Read-only	Id of the Peer in an HA setup.
sdxtools_public_key	<String>	Read-only	SDXTools HTTPS Public Key.
vm_description	<String>	Read-only	Description of the VM instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vm_rx	<Double>	Read-only	In Throughput of VM Instance in Mbps.
disk_allocation	<String>	Read-only	Disk allocation for VM Instance.
br_cpu_usage	<Double>	Read-only	CPU Usage (%) of Citrix SD-WAN Instance.
bandwidth_limit	<Double>	Read-only	bandwidth limit of Citrix SD-WAN Instance (K-Bits/sec).
sdxtools_version	<String>	Read-only	SDXTools version running on VM Instance.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- REBOOT
- STOP
- ADD
- FORCE_REBOOT
- GET (ALL)
- GET
- DELETE
- UPDATE
- FORCE_STOP
- START

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

REBOOT

URL: https://<MGMT-IP>/nitro/v2/config/br/id_value<String>?action=reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

STOP

URL: https://<MGMT-IP>/nitro/v2/config/br/id_value<String>?action=stop;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

ADD

URL: https://<MGMT-IP>/nitro/v2/config/br?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br": {

<b>"ip_address":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,
```

```
"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"plugin_ip_address":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"apa_netmask":<String_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,
```

```
"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,
```

```
"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"enable_quick_install":<Boolean_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,
```

```
"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"apa_gateway":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,
```

```
"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,
```

```
"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,
```

```
"vlan_1_8":<Integer_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"receiveuntagged_10_5":<Boolean_value>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,
```

```
"vrid_list_ipv4_10_2":<String_value[]>,

"apa_ip_address":<String_value>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt, "br":
[{

"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"bandwidth_mode":<String_value>,

"throughput_limit":<Double_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,
```

```
"disk_space":<Double_value>,

"is_active":<Boolean_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"qosstatus":<String_value>,

"plugin_ip_address":<String_value>,

"ent_bw_total":<Integer_value>,

"vcpu_config":<Integer_value>,

"wan_in":<Double_value>,

"preferred_cpu_socket":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"apa_netmask":<String_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,
```

```
"vm_state":<String_value>,

"uuid":<String_value>,

"connected_plugins":<String_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"ha_peer_state":<String_value>,

"lan_out_prev":<Double_value>,

"wan_in_prev":<Double_value>,

"bypass":<String_value>,

"type":<String_value>,

"throughput":<Double_value>,

"template_name":<String_value>,

"status":<String_value>,
```

```
"gateway":<String_value>,

"max_plugins":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"wan_out":<Double_value>,

"vm_tx":<Double_value>,

"number_of_scu":<Integer_value>,

"prev_poll_time":<Double_value>,

"unacl_connections":<String_value>,

"license":<String_value>,

"domain_name":<String_value>,

"acl_connections":<String_value>,

"vm_memory_free":<Double_value>,

"br_memory_usage":<Double_value>,

"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"image_name":<String_value>,

"hostname":<String_value>,

"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"vnc_console_url":<String_value>,
```

```
"wan_out_prev":<Double_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"lan_out":<Double_value>,

"user_driven":<Boolean_value>,

"active_connections":<Double_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"system_load":<Double_value>,

"is_managed":<Boolean_value>,
```

```
"apa_gateway":<String_value>,

"number_of_cpu":<Integer_value>,

"ha_state":<String_value>,

"nexthop_v6":<String_value>,

"operation_status":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"is_swg":<Boolean_value>,

"seq_no":<Double_value>,

"dns":<String_value>,

"data_reduction":<Double_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,
```

```
"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],
```

```
"last_updated_time":<Double_value>,

"port_model":<Integer_value>,

"uptime":<String_value>,

"ha_peer_ip_address":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"license_grace_time":<Integer_value>,

"havmip":<String_value>,

"la_mgmt":<Boolean_value>,

"lan_in_prev":<Double_value>,

"vlan_id_0_2":<Integer_value>,

"assigned_cpu_socket":<String_value>,

"number_of_ssl_cores":<Integer_value>,

"lan_in":<Double_value>,

"ha_peer_id":<Double_value>,

"sdxtools_public_key":<String_value>,

"user_profile_name":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,
```

```
"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"ip_address":<String_value>,

"br_cpu_usage":<Double_value>,

"throughput_allocation_mode":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"apa_ip_address":<String_value>,

"bandwidth_limit":<Double_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,
```

```
"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"skip_reboot":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,
```

```
"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,
```

```
"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,
```

```
"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,
```

```
"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"enable_quick_install":<Boolean_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

FORCE_REBOOT

URL: https://<MGMT-IP>/nitro/v2/config/br/id_value<String>?action=force_reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/br **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/br **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of br resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/br **?pagesize=#no;pageno=#no** Use this query-parameter to get the br resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/br **?count=yes** Use this query-parameter to get the count of br resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt, "br":
[{

"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"bandwidth_mode":<String_value>,

"throughput_limit":<Double_value>,
```

```
"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,

"disk_space":<Double_value>,

"is_active":<Boolean_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"qosstatus":<String_value>,

"plugin_ip_address":<String_value>,

"ent_bw_total":<Integer_value>,

"vcpu_config":<Integer_value>,

"wan_in":<Double_value>,

"preferred_cpu_socket":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"apa_netmask":<String_value>,
```

```
"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,

"vm_state":<String_value>,

"uuid":<String_value>,

"connected_plugins":<String_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"ha_peer_state":<String_value>,

"lan_out_prev":<Double_value>,

"wan_in_prev":<Double_value>,

"bypass":<String_value>,

"type":<String_value>,
```

```
"throughput":<Double_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"max_plugins":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"wan_out":<Double_value>,

"vm_tx":<Double_value>,

"number_of_scu":<Integer_value>,

"prev_poll_time":<Double_value>,

"unacl_connections":<String_value>,

"license":<String_value>,

"domain_name":<String_value>,

"acl_connections":<String_value>,

"vm_memory_free":<Double_value>,

"br_memory_usage":<Double_value>,

"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"image_name":<String_value>,

"hostname":<String_value>,

"is_internal_intf_added":<Boolean_value>,
```

```
"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"vnc_console_url":<String_value>,

"wan_out_prev":<Double_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"lan_out":<Double_value>,

"user_driven":<Boolean_value>,

"active_connections":<Double_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"host_id":<String_value>,
```

```
"has_sdxtools":<Boolean_value>,

"system_load":<Double_value>,

"is_managed":<Boolean_value>,

"apa_gateway":<String_value>,

"number_of_cpu":<Integer_value>,

"ha_state":<String_value>,

"nexthop_v6":<String_value>,

"operation_status":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"is_swg":<Boolean_value>,

"seq_no":<Double_value>,

"dns":<String_value>,

"data_reduction":<Double_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,
```

```
"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>
```

```
"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"last_updated_time":<Double_value>,

"port_model":<Integer_value>,

"uptime":<String_value>,

"ha_peer_ip_address":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"license_grace_time":<Integer_value>,

"havmip":<String_value>,

"la_mgmt":<Boolean_value>,

"lan_in_prev":<Double_value>,

"vlan_id_0_2":<Integer_value>,

"assigned_cpu_socket":<String_value>,

"number_of_ssl_cores":<Integer_value>,

"lan_in":<Double_value>,

"ha_peer_id":<Double_value>,

"sdxtools_public_key":<String_value>,

"user_profile_name":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,
```

```
"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"ip_address":<String_value>,

"br_cpu_usage":<Double_value>,

"throughput_allocation_mode":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"apa_ip_address":<String_value>,

"bandwidth_limit":<Double_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,
```

```
"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"skip_reboot":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,
```

```
"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,
```

```
"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,
```

```
"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,
```

```
"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"enable_quick_install":<Boolean_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/br/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt, "br":
[{
```

```
"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"bandwidth_mode":<String_value>,

"throughput_limit":<Double_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,

"disk_space":<Double_value>,

"is_active":<Boolean_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"qosstatus":<String_value>,

"plugin_ip_address":<String_value>,

"ent_bw_total":<Integer_value>,

"vcpu_config":<Integer_value>,

"wan_in":<Double_value>,

"preferred_cpu_socket":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,
```

```
"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"apa_netmask":<String_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,

"vm_state":<String_value>,

"uuid":<String_value>,

"connected_plugins":<String_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,
```

```
"ha_peer_state":<String_value>,

"lan_out_prev":<Double_value>,

"wan_in_prev":<Double_value>,

"bypass":<String_value>,

"type":<String_value>,

"throughput":<Double_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"max_plugins":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"wan_out":<Double_value>,

"vm_tx":<Double_value>,

"number_of_scu":<Integer_value>,

"prev_poll_time":<Double_value>,

"unacl_connections":<String_value>,

"license":<String_value>,

"domain_name":<String_value>,

"acl_connections":<String_value>,

"vm_memory_free":<Double_value>,

"br_memory_usage":<Double_value>,
```

```
"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"image_name":<String_value>,

"hostname":<String_value>,

"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"vnc_console_url":<String_value>,

"wan_out_prev":<Double_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"lan_out":<Double_value>,

"user_driven":<Boolean_value>,

"active_connections":<Double_value>,
```

```
"host_type":<String_value>,

"tenant_id":<String_value>,

"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"system_load":<Double_value>,

"is_managed":<Boolean_value>,

"apa_gateway":<String_value>,

"number_of_cpu":<Integer_value>,

"ha_state":<String_value>,

"nexthop_v6":<String_value>,

"operation_status":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"is_swg":<Boolean_value>,

"seq_no":<Double_value>,

"dns":<String_value>,

"data_reduction":<Double_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,
```

```
"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,
```

```
"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"last_updated_time":<Double_value>,

"port_model":<Integer_value>,

"uptime":<String_value>,

"ha_peer_ip_address":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"license_grace_time":<Integer_value>,

"havmip":<String_value>,

"la_mgmt":<Boolean_value>,

"lan_in_prev":<Double_value>,

"vlan_id_0_2":<Integer_value>,

"assigned_cpu_socket":<String_value>,

"number_of_ssl_cores":<Integer_value>,

"lan_in":<Double_value>,
```

```
"ha_peer_id":<Double_value>,

"sdxtools_public_key":<String_value>,

"user_profile_name":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"ip_address":<String_value>,

"br_cpu_usage":<Double_value>,

"throughput_allocation_mode":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"apa_ip_address":<String_value>,

"bandwidth_limit":<Double_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,
```

```
"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"skip_reboot":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,
```

```
"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,
```

```
"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,
```

```
"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,
```

```
"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"enable_quick_install":<Boolean_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/br/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/br/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br":{

<b>"id":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,
```

```
"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"plugin_ip_address":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"apa_netmask":<String_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,
```

```
"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,
```

```
"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"enable_quick_install":<Boolean_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,
```

```
"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"apa_gateway":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,
```

```
"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,
```

```
"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,
```

```
"vrid_list_ipv4_10_3":<String_value[]>,

"receiveuntagged_10_5":<Boolean_value>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,
```

```
"vrid_list_ipv4_10_2":<String_value[]>,

"apa_ip_address":<String_value>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt, "br":
[{

"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"bandwidth_mode":<String_value>,

"throughput_limit":<Double_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,
```

```
"disk_space":<Double_value>,

"is_active":<Boolean_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"qosstatus":<String_value>,

"plugin_ip_address":<String_value>,

"ent_bw_total":<Integer_value>,

"vcpu_config":<Integer_value>,

"wan_in":<Double_value>,

"preferred_cpu_socket":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"apa_netmask":<String_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,
```

```
"vm_state":<String_value>,

"uuid":<String_value>,

"connected_plugins":<String_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"ha_peer_state":<String_value>,

"lan_out_prev":<Double_value>,

"wan_in_prev":<Double_value>,

"bypass":<String_value>,

"type":<String_value>,

"throughput":<Double_value>,

"template_name":<String_value>,

"status":<String_value>,
```

```
"gateway":<String_value>,

"max_plugins":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"wan_out":<Double_value>,

"vm_tx":<Double_value>,

"number_of_scu":<Integer_value>,

"prev_poll_time":<Double_value>,

"unacl_connections":<String_value>,

"license":<String_value>,

"domain_name":<String_value>,

"acl_connections":<String_value>,

"vm_memory_free":<Double_value>,

"br_memory_usage":<Double_value>,

"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"image_name":<String_value>,

"hostname":<String_value>,

"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"vnc_console_url":<String_value>,
```

```
"wan_out_prev":<Double_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"lan_out":<Double_value>,

"user_driven":<Boolean_value>,

"active_connections":<Double_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"system_load":<Double_value>,

"is_managed":<Boolean_value>,
```

```
"apa_gateway":<String_value>,

"number_of_cpu":<Integer_value>,

"ha_state":<String_value>,

"nexthop_v6":<String_value>,

"operation_status":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"is_swg":<Boolean_value>,

"seq_no":<Double_value>,

"dns":<String_value>,

"data_reduction":<Double_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,
```

```
"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],
```

```
"last_updated_time":<Double_value>,

"port_model":<Integer_value>,

"uptime":<String_value>,

"ha_peer_ip_address":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"license_grace_time":<Integer_value>,

"havmip":<String_value>,

"la_mgmt":<Boolean_value>,

"lan_in_prev":<Double_value>,

"vlan_id_0_2":<Integer_value>,

"assigned_cpu_socket":<String_value>,

"number_of_ssl_cores":<Integer_value>,

"lan_in":<Double_value>,

"ha_peer_id":<Double_value>,

"sdxtools_public_key":<String_value>,

"user_profile_name":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,
```

```
"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"ip_address":<String_value>,

"br_cpu_usage":<Double_value>,

"throughput_allocation_mode":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"apa_ip_address":<String_value>,

"bandwidth_limit":<Double_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,
```

```
"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"skip_reboot":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,
```

```
"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,
```

```
"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,
```

```
"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,
```

```
"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"enable_quick_install":<Boolean_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

FORCE_STOP

URL: https://<MGMT-IP>/nitro/v2/config/br/id_value<String>?action=force_stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

START

URL: https://<MGMT-IP>/nitro/v2/config/br/id_value<String>?action=start;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

br_adduser

Configuration for Repeater User resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
password	<String>	Read-write	Password. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Username. Minimum length = 1 Maximum length = 128
privilege	<String>	Read-write	Privilege.
br_ip_address_arr	<String[]>	Read-write	List of BR IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- ADDUSER

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADDUSER

URL: https://<MGMT-IP>/nitro/v2/config/br_adduser?action=adduser;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_adduser": {

<b>"password":<String_value></b>,

<b>"name":<String_value></b>,

<b>"privilege":<String_value></b>,

<b>"br_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_adduser":[{

"password":<String_value>,

"name":<String_value>,

"privilege":<String_value>,

"br_ip_address_arr":<String_value>}]}
```

br_appflowconfig

Configuration for br_appflowconfig resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
tcp	<Boolean>	Read-write	tcp.
tcp_hdx	<Boolean>	Read-write	tcp_hdx.
wanOpt	<Boolean>	Read-write	wanOpt.
geo_support	<Boolean>	Read-write	Is this Citrix SD-WAN device configured to support GEO location..
hdx	<Boolean>	Read-write	hdx.
id	<String>	Read-write	BR IP Address. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
ip_address	<String>	Read-write	BR IP Address. Minimum length = 1 Maximum length = 64
appflowfeature	<Boolean>	Read-write	appflowfeature.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

br_configfile

Configuration for Repeater Configuration Files resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET

- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/br_configfile/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/br_configfile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/br_configfile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of br_configfile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/br_configfile **? pagesize=#no;pageno=#no** Use this query-parameter to get the br_configfile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/br_configfile **?count=yes** Use this query-parameter to get the count of br_configfile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_configfile":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/br_configfile/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_configfile":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

UPLOAD

URL: https://<MGMT-IP>/nitro/v2/upload/br_configfile **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

br_configurealert

Configuration for Configure Alert Settings resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
level	<String>	Read-write	Alert Level. Minimum length = 1 Maximum length = 128
name	<String>	Read-write	Alert name. Minimum length = 1 Maximum length = 128
br_ip_address_arr	<String[]>	Read-write	List of BR IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- CONFIGUREALERT

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

CONFIGUREALERT

URL: https://<MGMT-IP>/nitro/v2/config/br_configurealert?action=configurealert;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_configurealert": {

<b>"level":<String_value></b>,

<b>"name":<String_value></b>,

<b>"br_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_configurealert":[{

"level":<String_value>,

"name":<String_value>,

"br_ip_address_arr":<String_value>}]}
```

br_configurebandwidth5x

Configuration for Configure Bandwidth for Repeater devices of version 5x or earlier resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
bandwidth	<String>	Read-write	Bandwidth.
receive_limit	<Integer>	Read-write	Receive Limit.
min_rate	<Integer>	Read-write	Minimum bandwidth rate.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
boostmode	<String>	Read-write	Boost Mode. Minimum length = 1 Maximum length = 128
br_ip_address_arr	<String[]>	Read-write	List of BR IP Address. Minimum length = 1 Maximum length = 64
send_limit	<Integer>	Read-write	Send Limit.

Operations

(click to see Properties)

- CONFIGUREBANDWIDTH5X

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

CONFIGUREBANDWIDTH5X

URL: https://<MGMT-IP>/nitro/v2/config/br_configurebandwidth5x?action=configurebandwidth5x;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_configurebandwidth5x": {

<b>"bandwidth":<String_value></b>,

<b>"receive_limit":<Integer_value></b>,

<b>"min_rate":<Integer_value></b>,

<b>"boostmode":<String_value></b>,

<b>"br_ip_address_arr":<String_value[]></b>,

<b>"send_limit":<Integer_value></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_configurebandwidth5x":[{

"bandwidth":<String_value>,

"receive_limit":<Integer_value>,

"min_rate":<Integer_value>,

"boostmode":<String_value>,

"br_ip_address_arr":<String_value>,

"send_limit":<Integer_value>}]}
```

br_configurebandwidth6x

Configuration for Configure Bandwidth for Repeater devices of version 6x or later resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receive_limit	<Integer>	Read-write	Receive Limit.
boostmode	<String>	Read-write	Boost Mode. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
br_ip_address_arr	<String[]>	Read-write	List of BR IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- CONFIGUREBANDWIDTH6X

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

CONFIGUREBANDWIDTH6X

URL: https://<MGMT-IP>/nitro/v2/config/br_configurebandwidth6x?action=configurebandwidth6x;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_configurebandwidth6x": {

<b>"receive_limit":<Integer_value></b>,

<b>"boostmode":<String_value></b>,

<b>"br_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_configurebandwidth6x":[{

"receive_limit":<Integer_value>,

"boostmode":<String_value>,
```

```
"br_ip_address_arr":<String_value>}]}
```

br_configuresyslog

Configuration for Configure Syslog Server Settings resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
br_ip_address_arr	<String[]>	Read-write	List of BR IP Address. Minimum length = 1 Maximum length = 64
port	<Integer>	Read-write	Syslog server port number.
syslog_ip	<String>	Read-write	Syslog Server IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- CONFIGURESYSLOG

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

CONFIGURESYSLOG

URL: https://<MGMT-IP>/nitro/v2/config/br_configuresyslog?action=configuresyslog;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_configuresyslog": {

<b>"br_ip_address_arr":<String_value[]></b>,

<b>"port":<Integer_value></b>,

<b>"syslog_ip":<String_value></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_configuresyslog":[{

"br_ip_address_arr":<String_value>,

"port":<Integer_value>,

"syslog_ip":<String_value>}]}
```

br_currentconfig

Configuration for Repeater Current Configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
br_ip_address	<String>	Read-write	Repeater IP Address. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
response	<String>	Read-only	response.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/br_currentconfig **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_currentconfig":[{

"response":<String_value>,

"usefor_replication":<Boolean_value>,

"br_ip_address":<String_value>}]}
```

br_device_profile

Configuration for br_device_profile resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Profile Name. Minimum length = 1 Maximum length = 128
svm_ns_comm	<String>	Read-write	Communication protocol (http or https) with Instances. Minimum length = 1 Maximum length = 10
use_global_setting_for_communication_with_ns	<Boolean>	Read-write	True, if the communication with Instance needs to be global and not device specific.
id	<String>	Read-write	Id is system generated key for all the device profiles.
type	<String>	Read-write	Profile Type, This must be with in specified supported instance types: blx,sdvanvw,ns,nssdx,cbwanopt,cpx. Minimum length = 1 Maximum length = 128
ns_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix ADC profiles.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
password	<String>	Read-write	Instance credentials.Password for this profile. Minimum length = 1 Maximum length = 127
snmpsecuritylevel	<String>	Read-write	SNMP v3 security level for this profile.
username	<String>	Read-write	Instance credentials.Username provided in the profile will be used to contact the instance. Minimum length = 1 Maximum length = 127
snmpauthprotocol	<String>	Read-write	SNMP v3 auth protocol for this profile.
ssh_port	<String>	Read-write	SSH port to connect to the device.
snmpprivprotocol	<String>	Read-write	SNMP v3 priv protocol for this profile.
host_password	<String>	Read-write	Host Password for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
snmpversion	<String>	Read-write	SNMP version for this profile.
passphrase	<String>	Read-write	Passphrase with which private key is encrypted.
host_username	<String>	Read-write	Host User Name for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpsecurityname	<String>	Read-write	SNMP v3 security name for this profile. Maximum length = 31
ssl_private_key	<String>	Read-write	SSL Private Key for key based authentication.
ssl_cert	<String>	Read-write	SSL Certificate for certificate based authentication.
http_port	<Integer>	Read-write	HTTP port to connect to the device.
snmpcommunity	<String>	Read-write	SNMP community for this profile. Maximum length = 31

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
https_port	<Integer>	Read-write	HTTPS port to connect to the device.
max_wait_time_reboot	<String>	Read-write	Max waiting time to reboot Citrix ADC.
snmpprivpassword	<String>	Read-write	SNMP v3 priv password for this profile. Minimum length = 8 Maximum length = 31
cb_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix SD-WAN profiles.
snmpauthpassword	<String>	Read-write	SNMP v3 auth password for this profile. Minimum length = 8 Maximum length = 31
is_default	<Boolean>	Read-only	True, if this profile is system generated and can not be deleted.
tenant_id	<String>	Read-only	Tenant Id of device profile.

Operations

(click to see Properties)

- ADD

- DELETE
- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/br_device_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_device_profile": {

<b>"name":<String_value></b>,

<b>"password":<String_value></b>,

<b>"username":<String_value></b>,

"snmpsecuritylevel":<String_value>,

"svm_ns_comm":<String_value>,

"ssh_port":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"type":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"https_port":<Integer_value>,
```

```
"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"ns_profile_name":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,
```

```
"type":<String_value>,

"tenant_id":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,
```

```
"snmpauthpassword":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/br_device_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/br_device_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/br_device_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of br_device_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/br_device_profile **? pagesize=#no;pageno=#no** Use this query-parameter to get the br_device_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/br_device_profile **?count=yes** Use this query-parameter to get the count of br_device_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"tenant_id":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,
```

```
"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/br_device_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"tenant_id":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,
```

```
"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

br_disable

Configuration for Disable Repeater Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
br_ip_address_arr	<String[]>	Read-write	List of Repeater IP Address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/br_disable?action=disable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_disable": {

<b>"br_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_disable":[{

"br_ip_address_arr":<String_value>}]}
```

br_enable

Configuration for Enable Repeater Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
br_ip_address_arr	<String[]>	Read-write	List of Repeater IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- ENABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/br_enable?action=enable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_enable": {

<b>"br_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_enable":[{

"br_ip_address_arr":<String_value>}]}
```

br_image

Configuration for Repeater Build Files resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/br_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/br_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/br_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of br_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/br_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the br_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/br_image **?count=yes** Use this query-parameter to get the count of br_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_image":[{
```

```
"file_location_path":<String_value>,

"file_name":<String_value>,

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"is_dir":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/br_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_image":[{

"file_location_path":<String_value>,

"file_name":<String_value>,
```

```
"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"is_dir":<Boolean_value>}]}
```

UPLOAD

URL: https://<MGMT-IP>/nitro/v2/upload/br_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

br_locallicense

Configuration for Local License on Repeater Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
br_ip_address_arr	<String[]>	Read-write	List of Repeater IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- CONFIGURELICENSE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

CONFIGURELICENSE

URL: https://<MGMT-IP>/nitro/v2/config/br_locallicense?action=configurelicense;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_locallicense": {

<b>"br_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_locallicense":[{

"br_ip_address_arr":<String_value>}]}
```

br_remotelicense

Configuration for Remote License Server on Repeater Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
model	<String>	Read-write	License Server Model. Minimum length = 1 Maximum length = 128
br_ip_address_arr	<String[]>	Read-write	List of Repeater IP Address. Minimum length = 1 Maximum length = 64
port	<Integer>	Read-write	Remote License Server Port number.
server_ip	<String>	Read-write	Remote License Server IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- CONFIGUREREMOTELICENSE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

CONFIGUREREMOTELICENSE

URL: https://<MGMT-IP>/nitro/v2/config/br_remotelicense?action=configureremotelicense;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_remotelicense": {

<b>"model":<String_value></b>,

<b>"br_ip_address_arr":<String_value[]></b>,

<b>"port":<Integer_value></b>,

<b>"server_ip":<String_value></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_remotelicense":[{

"model":<String_value>,

"br_ip_address_arr":<String_value>,

"port":<Integer_value>,

"server_ip":<String_value>}]}
```

br_replicateconfig

Configuration for Repeater Replicate Configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
target_ip_address_arr	<String[]>	Read-write	Target Repeater IP Address Array.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
source_ip_address	<String>	Read-write	Source Repeater IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- REPLICATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

REPLICATE

URL: https://<MGMT-IP>/nitro/v2/config/br_replicateconfig?action=replicate;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_replicateconfig": {

"target_ip_address_arr":<String_value[]>,

"source_ip_address":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_replicateconfig":[{
```

```
"target_ip_address_arr":<String_value>,

"source_ip_address":<String_value>}]}
```

br_restart

Configuration for Restart Repeater Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
br_ip_address_arr	<String[]>	Read-write	List of Repeater IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- RESTART

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

RESTART

URL: https://<MGMT-IP>/nitro/v2/config/br_restart?action=restart;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_restart": {

<b>"br_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_restart":[{

"br_ip_address_arr":<String_value>}]}
```

br_restoreconfig

Configuration for Restore configuration of Repeater Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_name	<String>	Read-write	Repeater Config File Name. Minimum length = 1 Maximum length = 128
br_ip_address_arr	<String[]>	Read-write	List of Repeater IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- RESTORECONFIG

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

RESTORECONFIG

URL: https://<MGMT-IP>/nitro/v2/config/br_restoreconfig?action=restoreconfig;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_restoreconfig": {

<b>"file_name":<String_value></b>,

<b>"br_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_restoreconfig":[{

"file_name":<String_value>,

"br_ip_address_arr":<String_value>}]}
```

br_snmpmanager

Configuration for Add SNMP Manager resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
community	<String>	Read-write	SNMP Community. Minimum length = 1 Maximum length = 128
netmask	<String>	Read-write	SNMP Manager Netmask. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 50
manager_ip	<String>	Read-write	SNMP Manager IP. Minimum length = 1 Maximum length = 128
br_ip_address_arr	<String[]>	Read-write	List of BR IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- ADDMANAGER

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADDMANAGER

URL: https://<MGMT-IP>/nitro/v2/config/br_snmpmanager?action=addmanager;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_snmpmanager": {

<b>"community":<String_value></b>,

<b>"netmask":<String_value></b>,

<b>"manager_ip":<String_value></b>,

<b>"br_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_snmpmanager":[{

"community":<String_value>,

"netmask":<String_value>,

"manager_ip":<String_value>,

"br_ip_address_arr":<String_value>}]}
```

br_upgrade

Configuration for Upgrade Repeater Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	Repeater Image Name. Minimum length = 1 Maximum length = 128
br_ip_address_arr	<String[]>	Read-write	List of Repeater IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- UPGRADE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

UPGRADE

URL: https://<MGMT-IP>/nitro/v2/config/br_upgrade?action=upgrade;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_upgrade": {

<b>"image_name":<String_value></b>,

<b>"br_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_upgrade":[{

"mode":<Integer_value>,

"image_name":<String_value>,

"br_ip_address_arr":<String_value>}]}
```

Citrix-SD-WAN-Accelerator

- br_app_by_wan_volume
- br_broker
- br_broker_image
- br_broker_snmpmanager

- br_broker_upgrade

br_app_by_wan_volume

Configuration for Top Apps by wan volume resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address	<String>	Read-write	IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key for all apps..
app_name	<String>	Read-only	app name.
total_wan_volume	<Double>	Read-only	Total WAN volume for an app in Mbps.
total_received_wan_volume	<Double>	Read-only	Total Received WAN Volume for an app in Mbps.
total_sent_wan_volume	<Double>	Read-only	Total Sent WAN Volume for an app in Mbps.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/br_app_by_wan_volume **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/br_app_by_wan_volume **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of br_app_by_wan_volume resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/br_app_by_wan_volume **?pagesize=#no;pageno=#no** Use this query-parameter to get the br_app_by_wan_volume resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/br_app_by_wan_volume **?count=yes** Use this query-parameter to get the count of br_app_by_wan_volume resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_app_by_wan_volume":[{

"app_name":<String_value>,

"total_wan_volume":<Double_value>,

"ip_address":<String_value>,

"id":<String_value>,

"total_received_wan_volume":<Double_value>,

"total_sent_wan_volume":<Double_value>}]}
```

br_broker

Configuration for Citrix SD-WAN Accelerator Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
throughput_limit	<Double>	Read-write	Throughput Limit in Mbps set for VM Instance.
vlan_id_0_1	<Integer>	Read-write	VLAN id for the management interface 0/1. Maximum value =
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1 Maximum length = 64
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device. Minimum length = 1 Maximum length = 64
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.
host_ip_address	<String>	Read-write	Host IPAddress where VM is provisioned. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the managed devices.
ipv6_address	<String>	Read-write	IPv6 Address.
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
number_of_acu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
device_family	<String>	Read-write	Device Family. Minimum length = 1 Maximum length = 64
type	<String>	Read-write	Type of device, (Xen | NS). Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	Assign throughput in Mbps to VM Instance.
template_name	<String>	Read-write	Template Name, This parameter is used

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			while provisioning VM Instance with template, template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64
number_of_scu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
domain_name	<String>	Read-write	Domain name of VM Device. Minimum length = 1 Maximum length = 128
image_name	<String>	Read-write	Image Name, This parameter is used while provisioning VM Instance with XVA image, template_name is given priority if provided along with image_name. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256
vm_memory_total	<Double>	Read-write	Total Memory of VM Instance in MB. 2048MB, 5120MB.
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
description	<String>	Read-write	Description of managed device. Minimum length = 1 Maximum length = 512
ip_list	<String>	Read-write	Specify the list of Citrix SD-WAN IPs for which this device will act as a Citrix SD-WAN Accelerator.. Minimum length = 1 Maximum length = 1024
fips_partition_name	<String>	Read-write	FIPS Partition Name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 128
burst_priority	<Integer>	Read-write	Burst Priority of the VM Instance between 1 and 4.
is_managed	<Boolean>	Read-write	Is Managed.
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device. Minimum length = 1 Maximum length = 128
network_interfaces	<network_interface[]>	Read-write	Network Interfaces.
last_updated_time	<Double>	Read-write	Last Updated Time.
license_edition	<String>	Read-write	Edition of instance.
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..
la_mgmt	<Boolean>	Read-write	Bond consisting of management ports on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_id_0_2	<Integer>	Read-write	VLAN id for the management interface 0/2. Maximum value =
number_of_ssl_cores	<Integer>	Read-write	Assign number of ssl virtual functions to VM Instance.
max_burst_throughput	<Double>	Read-write	Maximum burst throughput in Mbps of VM Instance.
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal network (not required to be set for SDWAN).
node_id	<String>	Read-write	Node identification of a device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1 Maximum length = 64
throughput_allocation_mode	<Integer>	Read-write	Throughput Allocation Mode: 0-Fixed, 1-Burst-able.
number_of_cores	<Integer>	Read-write	Number of cores that are assigned to VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
std_bw_total	<Integer>	Read-write	Standard Bandwidth.
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be A-B, and for Cluster it will be CLIP. Minimum length = 1 Maximum length = 128
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1 Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.
vlan_1_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/2 on VM Instance. Maximum value =
vrid_list_ipv4_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV4 VMAC Generation.
vlan_1_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 1/7 on VM Instance. Maximum value =
vrid_list_ipv6_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV6 VMAC Generation.
vrid_list_ipv4_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV4 VMAC Generation.
vrid_list_ipv6_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV6 VMAC Generation.
if_10_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/4 on VM Instance.
vlan_10_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/5 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV6 VMAC Generation.
vrid_list_ipv6_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV6 VMAC Generation.
if_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/5 on VM Instance.
receiveuntagged_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/4 on VM Instance.
vrid_list_ipv6_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV6 VMAC Generation.
vrid_list_ipv6_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 1/3for IPV6 VMAC Generation.
if_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/3 on VM Instance.
vrid_list_ipv6_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV6 VMAC Generation.
receiveuntagged_10 _4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/4 on VM Instance.
vrid_list_ipv6_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV6 VMAC Generation.
vlan_10_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/6 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/5 on VM Instance.
is_new_crypto	<Boolean>	Read-write	True if number_of_acu/ number_of_scu are used, false if number_of_ssl_core s is used.
vrid_list_ipv6_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV6 VMAC Generation.
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
if_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/7 on VM Instance.
receiveuntagged_10 _8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/8 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV6 VMAC Generation.
receiveuntagged_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/6 on VM Instance.
vlan_1_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/4 on VM Instance. Maximum value =
if_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/2 on VM Instance.
receiveuntagged_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/5 on VM Instance.
receiveuntagged_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Untagged Packets on 10/6 on VM Instance.
receiveuntagged_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/1 on VM Instance.
vrid_list_ipv4_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV4 VMAC Generation.
vrid_list_ipv6_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV6 VMAC Generation.
vrid_list_ipv6_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV6 VMAC Generation.
vrid_list_ipv4_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/2 on VM Instance.
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
vrid_list_ipv4_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV4 VMAC Generation.
receiveuntagged_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/7 on VM Instance.
vlan_10_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/4 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/7 on VM Instance.
receiveuntagged_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/1 on VM Instance.
vrid_list_ipv6_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV6 VMAC Generation.
if_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/1 on VM Instance.
if_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/8 on VM Instance.
l2_enabled	<Boolean>	Read-write	L2mode status of VM Instance.
vlan_10_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 10/7 on VM Instance. Maximum value =
vlan_10_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/1 on VM Instance. Maximum value =
receiveuntagged_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/2 on VM Instance.
reboot_vm_on_cpu_change	<Boolean>	Read-write	Reboot VMs on CPU change during resource allocation.
vlan_1_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/6 on VM Instance. Maximum value =
vrid_list_ipv4_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV4 VMAC Generation.
receiveuntagged_10_7	<Boolean>	Read-write	This property is deprecated by

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			network_interfaces;lt br;gtReceive Untagged Packets on 10/7 on VM Instance.
vrid_list_ipv4_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV4 VMAC Generation.
vlan_1_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/1 on VM Instance. Maximum value =
vrid_list_ipv6_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV6 VMAC Generation.
vlan_1_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/8 on VM Instance. Maximum value =
vlan_10_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 10/2 on VM Instance. Maximum value =
receiveuntagged_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/5 on VM Instance.
vrid_list_ipv4_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV4 VMAC Generation.
vrid_list_ipv4_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV4 VMAC Generation.
vrid_list_ipv4_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV4 VMAC Generation.
receiveuntagged_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			on 1/8 on VM Instance.
vrid_list_ipv4_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV4 VMAC Generation.
vlan_1_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/3 on VM Instance. Maximum value =
receiveuntagged_10 _3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/3 on VM Instance.
if_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/6 on VM Instance.
if_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/6 on VM Instance.
vlan_10_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVLAN for Network 10/3 on VM Instance. Maximum value =
vlan_1_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/5 on VM Instance. Maximum value =
if_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/3 on VM Instance.
vrid_list_ipv4_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV4 VMAC Generation.
if_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/4 on VM Instance.
vrid_list_ipv4_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/1 on VM Instance.
vrid_list_ipv4_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV4 VMAC Generation.
vlan_10_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/8 on VM Instance. Maximum value =
receiveuntagged_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/3 on VM Instance.
crypto_change_requires_reboot	<Boolean>	Read-write	True if the current changes made by user requires a reboot of the VM else false.
vrid_list_ipv4_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			IPV4 VMAC Generation.
if_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/8 on VM Instance.
vrid_list_ipv4_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV4 VMAC Generation.
vrid_list_ipv6_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for IPV6 VMAC Generation.
if_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/2 on VM Instance.
vrid_list_ipv6_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV6 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
instance_state	<String>	Read-only	State of device, UP only if device accessible.
disk_space	<Double>	Read-only	Show Disk Space (GB) available to VM Instance.
cpu_core_pe	<String>	Read-only	Packet Engine cores assigned to VM Instance.
wan_in	<Double>	Read-only	WAN In of Citrix SD-WAN in Mbps.
preferred_cpu_socket	<String>	Read-only	Preferred CPU Socket.
device_uuid	<String>	Read-only	Device UUID.
version	<String>	Read-only	Device Version.
vm_memory_used	<Double>	Read-only	Memory Used (MB) by VM Instance.
sysid	<String>	Read-only	System ID.
vm_state	<String>	Read-only	State of Virtual Machine (Running | Halted).
uuid	<String>	Read-only	UUID of VM Instance generated by Hypervisor.
cpu_core_mgmt	<String>	Read-only	Management CPU cores assigned to VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ssl_virtual_functions	<String>	Read-only	SSL Virtual Functions assigned to VM Instance.
lan_out_prev	<Double>	Read-only	Lan out of Citrix SD-WAN Instance in Mbps.
wan_in_prev	<Double>	Read-only	WAN In of Citrix SD-WAN Instance in Mbps.
status	<String>	Read-only	Status of managed device.
systemname	<String>	Read-only	Device System Name.
mac_eth0	<String>	Read-only	MAC Address of eth0 on VM Instance.
wan_out	<Double>	Read-only	WAN Out of Citrix SD-WAN in Mbps.
non_accelerated_traffic	<Double>	Read-only	Non Accelerated Traffic of Citrix SD-WAN in Mbps.
vm_tx	<Double>	Read-only	Out Throughput of VM Instance in Mbps.
prev_poll_time	<Double>	Read-only	Previous time.
vm_memory_free	<Double>	Read-only	Free Memory (MB) available in VM Instance.
vm_memory_usage	<Double>	Read-only	Memory Usage (%) of VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vm_cpu_usage	<Double>	Read-only	CPU Usage (%) of VM Instance.
sdxtools_http_port	<Integer>	Read-only	Http Port number to communicate to SDXTools.
br_broker_cpu_usage	<Double>	Read-only	CPU Usage (%) of Citrix SD-WAN Accelerator Instance.
wan_out_prev	<Double>	Read-only	WAN out of Citrix SD-WAN Instance in Mbps.
reason	<String>	Read-only	Reason of failure for this managed device.
virtual_functions	<String>	Read-only	Virtual Functions assigned to VM Instance depending on SDX platform.
total_wan	<Double>	Read-only	Total WAN traffic in Mbps.
upsince	<String>	Read-only	Upsince of managed device.
domid	<Double>	Read-only	DOM Id assigned to VM Instance viewed in hypervisor.
lan_out	<Double>	Read-only	LAN Out of Citrix SD-WAN in Mbps.
active_connections	<Double>	Read-only	Total Active Connections on Citrix SD-WAN.
tenant_id	<String>	Read-only	Tenant ID.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
do_config	<Boolean>	Read-only	Do default config for managed device.
host_id	<String>	Read-only	Host ID.
has_sdxtools	<Boolean>	Read-only	True if SDX Tools are installed on this VM.
system_load	<Double>	Read-only	System Load (%) on Citrix SD-WAN.
number_of_cpu	<Integer>	Read-only	Number of CPU that is assigned to VM Instance.
is_swg	<Boolean>	Read-only	Boolean to indicate whether a VM is SWG.
non_accelerated_traffic_prev	<Double>	Read-only	Previous Non Accelerated Traffic of Citrix SD-WAN in Mbps.
total_lan	<Double>	Read-only	Total LAN traffic in Mbps.
data_reduction	<Double>	Read-only	Data Reduction (%).
uptime	<String>	Read-only	Uptime of device.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.
lan_in_prev	<Double>	Read-only	Lan In of Citrix SD-WAN Instance in Mbps.
assigned_cpu_socket	<String>	Read-only	Assigned CPU Socket 0 or 1.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
lan_in	<Double>	Read-only	Lan In of Citrix SD-WAN in Mbps.
sdxtools_public_key	<String>	Read-only	SDXTools HTTPS Public Key.
br_broker_memory_usage	<Double>	Read-only	Memory Usage (%) of Citrix SD-WAN Accelerator Instance.
vm_description	<String>	Read-only	Description of the VM instance.
vm_rx	<Double>	Read-only	In Throughput of VM Instance in Mbps.
disk_allocation	<String>	Read-only	Disk allocation for VM Instance.
sdxtools_version	<String>	Read-only	SDXTools version running on VM Instance.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- REBOOT
- STOP
- ADD
- FORCE_REBOOT
- GET (ALL)
- GET
- DELETE
- UPDATE
- FORCE_STOP

- START

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

REBOOT

URL: https://<MGMT-IP>/nitro/v2/config/br_broker/id_value<String>?action=reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_broker": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

STOP

URL: https://<MGMT-IP>/nitro/v2/config/br_broker/id_value<String>?action=stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_broker": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

ADD

URL: https://<MGMT-IP>/nitro/v2/config/br_broker?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_broker": {

<b>"ip_address":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,
```

```
"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,
```

```
"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,
```

```
"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,
```

```
"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,
```

```
"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,
```

```
"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,
```

```
"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"max_burst_throughput":<Double_value>,

"if_1_3":<Boolean_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,
```

```
"vrid_list_ipv4_10_2":<String_value[]>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_broker":[{

"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,

"disk_space":<Double_value>,

"cpu_core_pe":<String_value>,
```

```
"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vcpu_config":<Integer_value>,

"wan_in":<Double_value>,

"preferred_cpu_socket":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,

"vm_state":<String_value>,

"uuid":<String_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,
```

```
"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"lan_out_prev":<Double_value>,

"wan_in_prev":<Double_value>,

"type":<String_value>,

"throughput":<Double_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"wan_out":<Double_value>,

"non_accelerated_traffic":<Double_value>,

"vm_tx":<Double_value>,

"prev_poll_time":<Double_value>,

"number_of_scu":<Integer_value>,

"domain_name":<String_value>,
```

```
"vm_memory_free":<Double_value>,

"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"image_name":<String_value>,

"hostname":<String_value>,

"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"br_broker_cpu_usage":<Double_value>,

"vnc_console_url":<String_value>,

"wan_out_prev":<Double_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"total_wan":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,
```

```
"lan_out":<Double_value>,

"user_driven":<Boolean_value>,

"active_connections":<Double_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"system_load":<Double_value>,

"is_managed":<Boolean_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"is_swg":<Boolean_value>,

"seq_no":<Double_value>,

"dns":<String_value>,

"non_accelerated_traffic_prev":<Double_value>,

"total_lan":<Double_value>,

"data_reduction":<Double_value>,

"network_interfaces":[{
```

```
"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,
```

```
"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"last_updated_time":<Double_value>,

"uptime":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"lan_in_prev":<Double_value>,

"vlan_id_0_2":<Integer_value>,

"assigned_cpu_socket":<String_value>,

"number_of_ssl_cores":<Integer_value>,

"lan_in":<Double_value>,

"sdxtools_public_key":<String_value>,
```

```
"br_broker_memory_usage":<Double_value>,

"vm_description":<String_value>,

"user_profile_name":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"vm_rx":<Double_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"disk_allocation":<String_value>,

"privStr":<String_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"ping_state":<Integer_value>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,
```

```
"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"skip_reboot":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,
```

```
"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,
```

```
"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,
```

```
"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,
```

```
"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

FORCE_REBOOT

URL: https://<MGMT-IP>/nitro/v2/config/br_broker/id_value<String>?action=force_reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_broker": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/br_broker **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/br_broker **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of br_broker resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/br_broker **? pagesize=#no;pageno=#no** Use this query-parameter to get the br_broker resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/br_broker **?count=yes** Use this query-parameter to get the count of br_broker resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_broker":[{

"is_grace":<Boolean_value>,
```

```
"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,

"disk_space":<Double_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vcpu_config":<Integer_value>,

"wan_in":<Double_value>,

"preferred_cpu_socket":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"std_bw_available":<Integer_value>,
```

```
"internal_ip_address":<String_value>,

"sysid":<String_value>,

"vm_state":<String_value>,

"uuid":<String_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"lan_out_prev":<Double_value>,

"wan_in_prev":<Double_value>,

"type":<String_value>,

"throughput":<Double_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,
```

```
"systemname":<String_value>,

"mac_eth0":<String_value>,

"wan_out":<Double_value>,

"non_accelerated_traffic":<Double_value>,

"vm_tx":<Double_value>,

"prev_poll_time":<Double_value>,

"number_of_scu":<Integer_value>,

"domain_name":<String_value>,

"vm_memory_free":<Double_value>,

"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"image_name":<String_value>,

"hostname":<String_value>,

"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"br_broker_cpu_usage":<Double_value>,

"vnc_console_url":<String_value>,

"wan_out_prev":<Double_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,
```

```
"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"total_wan":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"lan_out":<Double_value>,

"user_driven":<Boolean_value>,

"active_connections":<Double_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"system_load":<Double_value>,

"is_managed":<Boolean_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,
```

```
"profile_name":<String_value>,

"is_swg":<Boolean_value>,

"seq_no":<Double_value>,

"dns":<String_value>,

"non_accelerated_traffic_prev":<Double_value>,

"total_lan":<Double_value>,

"data_reduction":<Double_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,
```

```
"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"last_updated_time":<Double_value>,

"uptime":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,
```

```
"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"lan_in_prev":<Double_value>,

"vlan_id_0_2":<Integer_value>,

"assigned_cpu_socket":<String_value>,

"number_of_ssl_cores":<Integer_value>,

"lan_in":<Double_value>,

"sdxtools_public_key":<String_value>,

"br_broker_memory_usage":<Double_value>,

"vm_description":<String_value>,

"user_profile_name":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"vm_rx":<Double_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"disk_allocation":<String_value>,

"privStr":<String_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"ping_state":<Integer_value>,
```

```
"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"skip_reboot":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,
```

```
"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,
```

```
"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,
```

```
"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,
```

```
"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/br_broker/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_broker":[{

"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,

"disk_space":<Double_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vcpu_config":<Integer_value>,

"wan_in":<Double_value>,

"preferred_cpu_socket":<String_value>,
```

```
"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,

"vm_state":<String_value>,

"uuid":<String_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,
```

```
"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"lan_out_prev":<Double_value>,

"wan_in_prev":<Double_value>,

"type":<String_value>,

"throughput":<Double_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"wan_out":<Double_value>,

"non_accelerated_traffic":<Double_value>,

"vm_tx":<Double_value>,

"prev_poll_time":<Double_value>,

"number_of_scu":<Integer_value>,

"domain_name":<String_value>,

"vm_memory_free":<Double_value>,

"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"image_name":<String_value>,

"hostname":<String_value>,
```

```
"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"br_broker_cpu_usage":<Double_value>,

"vnc_console_url":<String_value>,

"wan_out_prev":<Double_value>,

"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"total_wan":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"lan_out":<Double_value>,

"user_driven":<Boolean_value>,

"active_connections":<Double_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,
```

```
"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"system_load":<Double_value>,

"is_managed":<Boolean_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"is_swg":<Boolean_value>,

"seq_no":<Double_value>,

"dns":<String_value>,

"non_accelerated_traffic_prev":<Double_value>,

"total_lan":<Double_value>,

"data_reduction":<Double_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,
```

```
"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>
```

```
"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"last_updated_time":<Double_value>,

"uptime":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"lan_in_prev":<Double_value>,

"vlan_id_0_2":<Integer_value>,

"assigned_cpu_socket":<String_value>,

"number_of_ssl_cores":<Integer_value>,

"lan_in":<Double_value>,

"sdxtools_public_key":<String_value>,

"br_broker_memory_usage":<Double_value>,

"vm_description":<String_value>,

"user_profile_name":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,
```

```
"vm_rx":<Double_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"disk_allocation":<String_value>,

"privStr":<String_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"ping_state":<Integer_value>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,
```

```
"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"skip_reboot":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,
```

```
"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,
```

```
"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,
```

```
"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,
```

```
"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/br_broker/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/br_broker/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_broker":{

<b>"id":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,
```

```
"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,
```

```
"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,
```

```
"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,
```

```
"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,
```

```
"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,
```

```
"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"max_burst_throughput":<Double_value>,

"if_1_3":<Boolean_value>,
```

```
"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_broker":[{

"is_grace":<Boolean_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"name":<String_value>,

"disk_space":<Double_value>,

"cpu_core_pe":<String_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vcpu_config":<Integer_value>,

"wan_in":<Double_value>,

"preferred_cpu_socket":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"version":<String_value>,

"datacenter_id":<String_value>,

"old_max_burst_throughput":<Double_value>,
```

```
"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,

"vm_state":<String_value>,

"uuid":<String_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"lan_out_prev":<Double_value>,

"wan_in_prev":<Double_value>,

"type":<String_value>,

"throughput":<Double_value>,
```

```
"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"wan_out":<Double_value>,

"non_accelerated_traffic":<Double_value>,

"vm_tx":<Double_value>,

"prev_poll_time":<Double_value>,

"number_of_scu":<Integer_value>,

"domain_name":<String_value>,

"vm_memory_free":<Double_value>,

"vm_memory_usage":<Double_value>,

"vm_cpu_usage":<Double_value>,

"image_name":<String_value>,

"hostname":<String_value>,

"is_internal_intf_added":<Boolean_value>,

"vm_memory_total":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"br_broker_cpu_usage":<Double_value>,

"vnc_console_url":<String_value>,

"wan_out_prev":<Double_value>,
```

```
"provision_with_xva":<Boolean_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"ip_list":<String_value>,

"total_wan":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"lan_out":<Double_value>,

"user_driven":<Boolean_value>,

"active_connections":<Double_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"burst_priority":<Integer_value>,

"do_config":<Boolean_value>,

"host_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"system_load":<Double_value>,

"is_managed":<Boolean_value>,
```

```
"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"is_swg":<Boolean_value>,

"seq_no":<Double_value>,

"dns":<String_value>,

"non_accelerated_traffic_prev":<Double_value>,

"total_lan":<Double_value>,

"data_reduction":<Double_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,
```

```
"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"last_updated_time":<Double_value>,
```

```
"uptime":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"lan_in_prev":<Double_value>,

"vlan_id_0_2":<Integer_value>,

"assigned_cpu_socket":<String_value>,

"number_of_ssl_cores":<Integer_value>,

"lan_in":<Double_value>,

"sdxtools_public_key":<String_value>,

"br_broker_memory_usage":<Double_value>,

"vm_description":<String_value>,

"user_profile_name":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"vm_rx":<Double_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"disk_allocation":<String_value>,

"privStr":<String_value>,

"ip_address":<String_value>,
```

```
"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"ping_state":<Integer_value>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"sdxtools_version":<String_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,
```

```
"skip_reboot":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,
```

```
"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,
```

```
"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,
```

```
"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

FORCE_STOP

URL: https://<MGMT-IP>/nitro/v2/config/br_broker/id_value<String>?action=force_stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_broker": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

START

URL: https://<MGMT-IP>/nitro/v2/config/br_broker/id_value<String>?action=start;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_broker": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

br_broker_image

Configuration for Unified Repeater Build Files resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE

- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/br_broker_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/br_broker_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/br_broker_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of br_broker_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/br_broker_image **?pagesize=#no;pageno=#no** Use this query-parameter to get the br_broker_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/br_broker_image **?count=yes** Use this query-parameter to get the count of br_broker_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_broker_image":[{

"file_location_path":<String_value>,

"file_name":<String_value>,

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"is_dir":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/br_broker_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_broker_image":[{

"file_location_path":<String_value>,

"file_name":<String_value>,

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"is_dir":<Boolean_value>}]}
```

UPLOAD

URL: https://<MGMT-IP>/nitro/v2/upload/br_broker_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

br_broker_snmpmanager

Configuration for Add SNMP Manager resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
community	<String>	Read-write	SNMP Community. Minimum length = 1 Maximum length = 128
netmask	<String>	Read-write	SNMP Manager Netmask. Minimum length = 1 Maximum length = 50
manager_ip	<String>	Read-write	SNMP Manager IP. Minimum length = 1 Maximum length = 128
br_broker_ip_address_arr	<String[]>	Read-write	List of BR IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- ADDMANAGER

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADDMANAGER

URL: https://<MGMT-IP>/nitro/v2/config/br_broker_snmpmanager?action=addmanager;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_broker_snmpmanager": {

<b>"community":<String_value></b>,

<b>"netmask":<String_value></b>,

<b>"manager_ip":<String_value></b>,

<b>"br_broker_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_broker_snmpmanager":[{

"community":<String_value>,

"netmask":<String_value>,

"manager_ip":<String_value>,

"br_broker_ip_address_arr":<String_value>}]}
```

br_broker_upgrade

Configuration for Upgrade Unified Repeater Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	Unified Repeater Image Name. Minimum length = 1 Maximum length = 128
br_broker_ip_address_arr	<String[]>	Read-write	List of Unified Repeater IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- UPGRADE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

UPGRADE

URL: https://<MGMT-IP>/nitro/v2/config/br_broker_upgrade?action=upgrade;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"br_broker_upgrade": {

<b>"image_name":<String_value></b>,

<b>"br_broker_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"br_broker_upgrade":[{

"mode":<Integer_value>,

"image_name":<String_value>,

"br_broker_ip_address_arr":<String_value>}]}
```

Common-Resources

- device_generic_resrc
- device_group
- device_license_info
- device_profile
- devicewise_detail_summary
- host_device
- image
- inventory
- inventory_status
- managed_device
- network_interface
- ping
- traceroute
- vm_device

device_generic_resrc

Configuration for Device generic utility resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
failure_action	<String>	Read-write	Action taken on failure .
sync	<Boolean>	Read-write	Ping SVM to check its status.
payload	<String>	Read-write	Payload/Body Request .
device_family_type	<String>	Read-write	Device family type. Minimum length = 1
ipaddress	<String[]>	Read-write	List of SVM IP Address. Minimum length = 1 Maximum length = 64
resource_type	<String>	Read-write	Type of resource. Minimum length = 1
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- ADD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/device_generic_resrc?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"device_generic_resrc": {

<b>"payload":<String_value></b>,

<b>"device_family_type":<String_value></b>,

<b>"ipaddress":<String_value[]></b>,

<b>"resource_type":<String_value></b>,

"failure_action":<String_value>,

"sync":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_generic_resrc":[{

"act_id":<String_value>,

"failure_action":<String_value>,

"sync":<Boolean_value>,

"payload":<String_value>,

"device_family_type":<String_value>,

"ipaddress":<String_value>,

"resource_type":<String_value>}]}
```

device_group

Configuration for Device Group resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
static_device_list	<String>	Read-write	Devices in the group.
device_family	<String>	Read-write	Device Family. Minimum length = 1 Maximum length = 64
lock_acquiring_device	<String>	Read-write	Upgrade Lock acquiring device. Maximum length = 255
name	<String>	Read-write	Device Group Name. Maximum length = 255
duration	<Integer>	Read-write	Duration of Maintenance window for groups of category upgrade. Minimum value = 1 Maximum value =
maintenance_window_start	<String>	Read-write	Maintenance window start time for groups of category upgrade. Maximum length = 255
criteria_type	<String>	Read-write	Device Group Criteria. Maximum length = 255

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
upgrade_version	<String>	Read-write	New Available upgrade version for devices. Maximum length = 255
criteria_condn	<String>	Read-write	Tenant. Maximum length = 255
upgrade_lock	<Boolean>	Read-write	Lock to be acquired before upgrading device group.
id	<String>	Read-write	Id is system generated key for agent registered with NMX Cloud..
category	<String>	Read-write	Device group category. Will be default/upgrade.. Maximum length = 255
disable_upgrade	<Boolean>	Read-write	Setting to disable agent upgrades.
lock_acquire_time	<String>	Read-write	Upgrade Lock acquiring time. Maximum length = 255
criteria_value	<String>	Read-write	Criteria Value. Maximum length = 255
static_device_list_arr	<String[]>	Read-write	Static Device List.
tenant_id	<String>	Read-only	Tenant Id.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/device_group?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"device_group": {

<b>"duration":<Integer_value></b>,

"static_device_list":<String_value>,

"lock_acquiring_device":<String_value>,

"criteria_condn":<String_value>,

"category":<String_value>,

"id":<String_value>,

"criteria_value":<String_value>,

"device_family":<String_value>,

"name":<String_value>,

"upgrade_lock":<Boolean_value>,

"disable_upgrade":<Boolean_value>,

"lock_acquire_time":<String_value>,
```

```
"upgrade_version":<String_value>,

"static_device_list_arr":<String_value[]>,

"criteria_type":<String_value>,

"maintenance_window_start":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_group":[{

"static_device_list":<String_value>,

"device_family":<String_value>,

"lock_acquiring_device":<String_value>,

"name":<String_value>,

"duration":<Integer_value>,

"maintenance_window_start":<String_value>,

"criteria_type":<String_value>,

"upgrade_version":<String_value>,

"criteria_condn":<String_value>,

"upgrade_lock":<Boolean_value>,

"tenant_id":<String_value>,

"id":<String_value>,
```

```
"category":<String_value>,

"disable_upgrade":<Boolean_value>,

"lock_acquire_time":<String_value>,

"criteria_value":<String_value>,

"static_device_list_arr":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/device_group/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/device_group **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/device_group **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of device_group resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/device_group **?pagesize=#no;pageno=#no** Use this query-parameter to get the device_group resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/device_group **?count=yes** Use this query-parameter to get the count of device_group resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_group":[{

"static_device_list":<String_value>,

"device_family":<String_value>,

"lock_acquiring_device":<String_value>,

"name":<String_value>,

"duration":<Integer_value>,

"maintenance_window_start":<String_value>,
```

```
"criteria_type":<String_value>,

"upgrade_version":<String_value>,

"criteria_condn":<String_value>,

"upgrade_lock":<Boolean_value>,

"tenant_id":<String_value>,

"id":<String_value>,

"category":<String_value>,

"disable_upgrade":<Boolean_value>,

"lock_acquire_time":<String_value>,

"criteria_value":<String_value>,

"static_device_list_arr":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/device_group/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_group":[{

"static_device_list":<String_value>,

"device_family":<String_value>,

"lock_acquiring_device":<String_value>,

"name":<String_value>,

"duration":<Integer_value>,

"maintenance_window_start":<String_value>,

"criteria_type":<String_value>,

"upgrade_version":<String_value>,

"criteria_condn":<String_value>,

"upgrade_lock":<Boolean_value>,

"tenant_id":<String_value>,

"id":<String_value>,

"category":<String_value>,

"disable_upgrade":<Boolean_value>,

"lock_acquire_time":<String_value>,

"criteria_value":<String_value>,

"static_device_list_arr":<String_value>}]}
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/device_group/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"device_group":{

<b>"duration":<Integer_value></b>,

"static_device_list":<String_value>,

"lock_acquiring_device":<String_value>,

"criteria_condn":<String_value>,

"category":<String_value>,

"id":<String_value>,

"criteria_value":<String_value>,

"device_family":<String_value>,

"name":<String_value>,

"upgrade_lock":<Boolean_value>,

"disable_upgrade":<Boolean_value>,

"lock_acquire_time":<String_value>,

"upgrade_version":<String_value>,

"static_device_list_arr":<String_value[]>,

"criteria_type":<String_value>,
```

```
"maintenance_window_start":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_group":[{

"static_device_list":<String_value>,

"device_family":<String_value>,

"lock_acquiring_device":<String_value>,

"name":<String_value>,

"duration":<Integer_value>,

"maintenance_window_start":<String_value>,

"criteria_type":<String_value>,

"upgrade_version":<String_value>,

"criteria_condn":<String_value>,

"upgrade_lock":<Boolean_value>,

"tenant_id":<String_value>,

"id":<String_value>,

"category":<String_value>,

"disable_upgrade":<Boolean_value>,

"lock_acquire_time":<String_value>,
```

```
"criteria_value":<String_value>,

"static_device_list_arr":<String_value>}]}
```

device_license_info

Configuration for Device License Info resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
instance_mode	<String>	Read-write	instance mode of device.
details	<String>	Read-write	Details about license state.
last_updated_time	<Double>	Read-write	Last Updated Time.
configured_on_vm	<Integer>	Read-only	Configured license capacity of VM.
current_throughput	<String>	Read-only	Current Througput of device.
license_pool	<String>	Read-only	License Pool.
allocated_on_vm	<Integer>	Read-only	Running license capacity of VM.
name	<String>	Read-only	Name of managed device.
grace	<Boolean>	Read-only	True if this device is managed by NMX.
managed_device_id	<String>	Read-only	Device id managaed device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
license_state	<String>	Read-only	State of device, UP only if device accessible.
allocated	<Integer>	Read-only	Allocate license on License server.
hours_to_expiry	<Integer>	Read-only	No of Hours after which grace license expires.
ip_address	<String>	Read-only	IP Address for this managed device.
type	<String>	Read-only	Type of device, (Xen | NS).

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/device_license_info **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/device_license_info **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of device_license_info resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/device_license_info **? pagesize=#no;pageno=#no** Use this query-parameter to get the device_license_info resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/device_license_info **?count=yes** Use this query-parameter to get the count of device_license_info resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_license_info":[{

"configured_on_vm":<Integer_value>,

"current_throughput":<String_value>,

"license_pool":<String_value>,

"allocated_on_vm":<Integer_value>,

"name":<String_value>,

"instance_mode":<String_value>,

"details":<String_value>,

"grace":<Boolean_value>,

"managed_device_id":<String_value>,

"license_state":<String_value>,

"last_updated_time":<Double_value>,

"allocated":<Integer_value>,

"hours_to_expiry":<Integer_value>,

"ip_address":<String_value>,

"type":<String_value>}]}
```

device_profile

Configuration for Device Profile resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
svm_ns_comm	<String>	Read-write	Communication protocol (http or https) with Instances. Minimum length = 1 Maximum length = 10
name	<String>	Read-write	Profile Name. Minimum length = 1 Maximum length = 128
type	<String>	Read-write	Profile Type, This must be with in specified supported instance types: blx,sdvanvw,ns,nssdx,cbwanopt,cpx. Minimum length = 1 Maximum length = 128
id	<String>	Read-write	Id is system generated key for all the device profiles.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
use_global_setting_for_communication_with_ns	<Boolean>	Read-write	True, if the communication with Instance needs to be global and not device specific.
ssl_private_key	<String>	Read-write	SSL Private Key for key based authentication.
snmpauthprotocol	<String>	Read-write	SNMP v3 auth protocol for this profile.
snmpsecurityname	<String>	Read-write	SNMP v3 security name for this profile. Maximum length = 31
ssl_cert	<String>	Read-write	SSL Certificate for certificate based authentication.
http_port	<Integer>	Read-write	HTTP port to connect to the device.
ns_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix ADC profiles.
ssh_port	<String>	Read-write	SSH port to connect to the device.
password	<String>	Read-write	Instance credentials.Password for this profile. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 127
snmpsecuritylevel	<String>	Read-write	SNMP v3 security level for this profile.
snmpcommunity	<String>	Read-write	SNMP community for this profile. Maximum length = 31
passphrase	<String>	Read-write	Passphrase with which private key is encrypted.
snmpprivprotocol	<String>	Read-write	SNMP v3 priv protocol for this profile.
https_port	<Integer>	Read-write	HTTPS port to connect to the device.
username	<String>	Read-write	Instance credentials.Username provided in the profile will be used to contact the instance. Minimum length = 1 Maximum length = 127
host_password	<String>	Read-write	Host Password for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
max_wait_time_reboot	<String>	Read-write	Max waiting time to reboot Citrix ADC.
snmpprivpassword	<String>	Read-write	SNMP v3 priv password for this profile. Minimum length = 8 Maximum length = 31
snmpversion	<String>	Read-write	SNMP version for this profile.
host_username	<String>	Read-write	Host User Name for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpauthpassword	<String>	Read-write	SNMP v3 auth password for this profile. Minimum length = 8 Maximum length = 31
cb_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix SD-WAN profiles.
tenant_id	<String>	Read-only	Tenant Id of device profile.
is_default	<Boolean>	Read-only	True, if this profile is system generated

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			and can not be deleted.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/device_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"device_profile": {

<b>"name":<String_value></b>,

<b>"password":<String_value></b>,

<b>"username":<String_value></b>,

"snmpsecurityname":<String_value>,

"snmpauthprotocol":<String_value>,

"ssl_private_key":<String_value>,

"svm_ns_comm":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,
```

```
"ns_profile_name":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"ssh_port":<String_value>,

"snmpsecuritylevel":<String_value>,

"snmpcommunity":<String_value>,

"passphrase":<String_value>,

"id":<String_value>,

"snmpprivprotocol":<String_value>,

"https_port":<Integer_value>,

"host_password":<String_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"type":<String_value>,

"snmpversion":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>,

"host_username":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_profile":[{

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"is_default":<Boolean_value>,

"name":<String_value>,

"type":<String_value>,

"id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"ssl_private_key":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"ns_profile_name":<String_value>,

"ssh_port":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"snmpcommunity":<String_value>,

"passphrase":<String_value>,

"snmpprivprotocol":<String_value>,

"https_port":<Integer_value>,
```

```
"username":<String_value>,

"host_password":<String_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"snmpversion":<String_value>,

"host_username":<String_value>,

"snmpauthpassword":<String_value>,

"cb_profile_name":<String_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/device_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/device_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/device_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of device_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/device_profile **? pagesize=#no;pageno=#no** Use this query-parameter to get the device_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/device_profile **?count=yes** Use this query-parameter to get the count of device_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_profile":[{

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"is_default":<Boolean_value>,

"name":<String_value>,
```

```
"type":<String_value>,

"id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"ssl_private_key":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"ns_profile_name":<String_value>,

"ssh_port":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"snmpcommunity":<String_value>,

"passphrase":<String_value>,

"snmpprivprotocol":<String_value>,

"https_port":<Integer_value>,

"username":<String_value>,

"host_password":<String_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"snmpversion":<String_value>,

"host_username":<String_value>,
```

```
"snmpauthpassword":<String_value>,

"cb_profile_name":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/device_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_profile":[{

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"is_default":<Boolean_value>,

"name":<String_value>,

"type":<String_value>,

"id":<String_value>,
```

```
"use_global_setting_for_communication_with_ns":<Boolean_value>,

"ssl_private_key":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"ns_profile_name":<String_value>,

"ssh_port":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"snmpcommunity":<String_value>,

"passphrase":<String_value>,

"snmpprivprotocol":<String_value>,

"https_port":<Integer_value>,

"username":<String_value>,

"host_password":<String_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"snmpversion":<String_value>,

"host_username":<String_value>,

"snmpauthpassword":<String_value>,

"cb_profile_name":<String_value>}]}
```

UPDATE

URL: https://<MGMT-IP>/nitro/v2/config/device_profile/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"device_profile":{

<b>"id":<String_value></b>,

"snmpsecurityname":<String_value>,

"snmpauthprotocol":<String_value>,

"ssl_private_key":<String_value>,

"svm_ns_comm":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"ns_profile_name":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"ssh_port":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"snmpcommunity":<String_value>,
```

```
"passphrase":<String_value>,

"name":<String_value>,

"snmpprivprotocol":<String_value>,

"https_port":<Integer_value>,

"username":<String_value>,

"host_password":<String_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"type":<String_value>,

"snmpversion":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>,

"host_username":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_profile":[{

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"is_default":<Boolean_value>,
```

```
"name":<String_value>,

"type":<String_value>,

"id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"ssl_private_key":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"ns_profile_name":<String_value>,

"ssh_port":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"snmpcommunity":<String_value>,

"passphrase":<String_value>,

"snmpprivprotocol":<String_value>,

"https_port":<Integer_value>,

"username":<String_value>,

"host_password":<String_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"snmpversion":<String_value>,
```

```
"host_username":<String_value>,

"snmpauthpassword":<String_value>,

"cb_profile_name":<String_value>}]}
```

devicewise_detail_summary

Configuration for Ns event devices summary resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_agent	<Boolean>	Read-write	Is this device acting as Gateway..
memory_usage	<Double>	Read-write	memory usage of a device.
cityname	<String>	Read-write	City Name. Maximum length = 255
agent_name	<String>	Read-write	Agent Name. Maximum length = 255
agent_id	<String>	Read-write	Agent Id. Maximum length = 255
throughput	<Double>	Read-write	Assign throughput in Mbps to VM Instance.
type	<String>	Read-write	Type of device, (Xen | NS).

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
instances	<String>	Read-only	Total number of devices managed by this agent.
agent_ip_address	<String>	Read-only	Total number of devices deployed city wise...
total_events	<Integer>	Read-only	Total minor events..
tenant_id	<String>	Read-only	Tenant Id.
platform	<String>	Read-only	Platform on which Agent is deployed. e.g. XenServer, kubernetes.
vm_cpu_usage	<Double>	Read-only	CPU Usage (%) of VM Instance.
ip_address	<String>	Read-only	Total number of devices deployed city wise...
instance_state	<String>	Read-only	State of device, UP only if device accessible.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/devicewise_detail_summary **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/devicewise_detail_summary **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of devicewise_detail_summary resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/devicewise_detail_summary **?pagesize=#no;pageno=#no** Use this query-parameter to get the devicewise_detail_summary resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/devicewise_detail_summary **?count=yes** Use this query-parameter to get the count of devicewise_detail_summary resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"devicewise_detail_summary":[{

"instances":<String_value>,

"is_agent":<Boolean_value>,

"memory_usage":<Double_value>,

"agent_ip_address":<String_value>,

"total_events":<Integer_value>,

"cityname":<String_value>,
```

```
"tenant_id":<String_value>,

"agent_name":<String_value>,

"agent_id":<String_value>,

"platform":<String_value>,

"vm_cpu_usage":<Double_value>,

"throughput":<Double_value>,

"ip_address":<String_value>,

"instance_state":<String_value>,

"type":<String_value>}]}
```

host_device

Configuration for host resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128
description	<String>	Read-write	Description of managed device. Minimum length = 1 Maximum length = 512
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
is_managed	<Boolean>	Read-write	Is Managed.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device. Minimum length = 1 Maximum length = 128
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
last_updated_time	<Double>	Read-write	Last Updated Time.
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.
id	<String>	Read-write	Id is system generated key for all the managed devices.
ipv6_address	<String>	Read-write	IPv6 Address.
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
license_edition	<String>	Read-write	Edition of instance.
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
device_family	<String>	Read-write	Device Family. Minimum length = 1 Maximum length = 64
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..
type	<String>	Read-write	Type of device, (Xen | NS). Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal network (not required to be set for SDWAN).
node_id	<String>	Read-write	Node identification of a device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1 Maximum length = 64
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be A-B, and for Cluster it will be CLIP. Minimum length = 1 Maximum length = 128
std_bw_total	<Integer>	Read-write	Standard Bandwidth.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1 Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
acu_total	<Integer>	Read-only	Total Asymmetric Crypto Units available in host.
instance_state	<String>	Read-only	State of device, UP only if device accessible.
enabled	<Boolean>	Read-only	host State.
reason	<String>	Read-only	Reason of failure for this managed device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
disk_space	<Double>	Read-only	Show Disk Space (MB) available on host.
disk_free	<Double>	Read-only	Show free Disk Space (MB) available on host.
upsince	<String>	Read-only	Upsince of managed device.
disk_usage	<Double>	Read-only	Disk usage (MB) for host.
tenant_id	<String>	Read-only	Tenant ID.
device_uuid	<String>	Read-only	Device UUID.
do_config	<Boolean>	Read-only	Do default config for managed device.
tx	<Double>	Read-only	Out Throughput (Mbps) of host.
version	<String>	Read-only	Device Version.
host_id	<String>	Read-only	Host ID.
number_of_cpu	<Integer>	Read-only	Number of total CPU of host.
memory_free	<Double>	Read-only	Free memory available (MB) in host.
sysid	<String>	Read-only	System ID.
uuid	<String>	Read-only	UUID of host.
scu_total	<Integer>	Read-only	Total Symmetric Crypto Units available in host.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
uptime	<String>	Read-only	Uptime of device.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.
scu_free	<Integer>	Read-only	Free Symmetric Crypto Units.
total_crypto_devices	<Integer>	Read-only	Total Crypto Devices.
cpu_usage	<Double>	Read-only	CPU Usage (%) of host.
acu_free	<Integer>	Read-only	Free Asymmetric Crypto Units.
memory_total	<Double>	Read-only	Total memory of host in MB.
status	<String>	Read-only	Status of managed device.
systemname	<String>	Read-only	Device System Name.
ssl_cores_total	<Integer>	Read-only	Total SSL Cores available in host.
rx	<Double>	Read-only	In Throughput (Mbps) of host.
memory_usage	<Double>	Read-only	Memory Usage (%) of host.
number_of_cpu_cores	<Integer>	Read-only	Number of total CPU cores of host.
act_id	<String>	Read-only	Activity Id.
ssl_cores_free	<Integer>	Read-only	Free SSL Cores.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/host_device **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/host_device **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of host_device resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/host_device **? pagesize=#no;pageno=#no** Use this query-parameter to get the host_device resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/host_device **?count=yes** Use this query-parameter to get the count of host_device resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"host_device":[{

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"acu_total":<Integer_value>,
```

```
"gateway_ipv6":<String_value>,

"instance_state":<String_value>,

"enabled":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"disk_free":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"disk_usage":<Double_value>,

"ent_bw_total":<Integer_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"tx":<Double_value>,
```

```
"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"is_managed":<Boolean_value>,

"instance_mode":<String_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"memory_free":<Double_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,

"uuid":<String_value>,

"scu_total":<Integer_value>,

"last_updated_time":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,
```

```
"license_edition":<String_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"scu_free":<Integer_value>,

"total_crypto_devices":<Integer_value>,

"cpu_usage":<Double_value>,

"acu_free":<Integer_value>,

"type":<String_value>,

"memory_total":<Double_value>,

"status":<String_value>,

"gateway":<String_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"ping_state":<Integer_value>,

"ssl_cores_total":<Integer_value>,

"rx":<Double_value>,

"memory_usage":<Double_value>,
```

```
"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"number_of_cpu_cores":<Integer_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"parent_inventory":<Boolean_value>,

"act_id":<String_value>,

"profile_username":<String_value>,

"default_password":<String_value>,

"profile_password":<String_value>,

"sync_operation":<Boolean_value>,

"ssl_cores_free":<Integer_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/host_device/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"host_device":[{

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"acu_total":<Integer_value>,

"gateway_ipv6":<String_value>,

"instance_state":<String_value>,

"enabled":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"disk_free":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"disk_usage":<Double_value>,
```

```
"ent_bw_total":<Integer_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"tx":<Double_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"is_managed":<Boolean_value>,

"instance_mode":<String_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"memory_free":<Double_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,
```

```
"uuid":<String_value>,

"scu_total":<Integer_value>,

"last_updated_time":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"scu_free":<Integer_value>,

"total_crypto_devices":<Integer_value>,

"cpu_usage":<Double_value>,

"acu_free":<Integer_value>,

"type":<String_value>,

"memory_total":<Double_value>,

"status":<String_value>,

"gateway":<String_value>,

"config_type":<Integer_value>,
```

```
"systemname":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"ping_state":<Integer_value>,

"ssl_cores_total":<Integer_value>,

"rx":<Double_value>,

"memory_usage":<Double_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"number_of_cpu_cores":<Integer_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"parent_inventory":<Boolean_value>,

"act_id":<String_value>,

"profile_username":<String_value>,

"default_password":<String_value>,

"profile_password":<String_value>,

"sync_operation":<Boolean_value>,

"ssl_cores_free":<Integer_value>}]}
```

image

Configuration for Image File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

inventory

Configuration for Inventory resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
device_ipaddress	<String>	Read-write	Device IP Address. Minimum length = 1 Maximum length = 64
agent_id	<String>	Read-write	Agent ID. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
devices	<String[]>	Read-write	List of Device IPAddresses.
device_type	<String>	Read-only	Status of inventory.
act_id	<String>	Read-only	Activity Id.
inventory_status	<String>	Read-only	Status of inventory.

Operations

(click to see Properties)

- ADD
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/inventory?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"inventory": {

"devices":<String_value[]>,

"device_ipaddress":<String_value>,

"agent_id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"inventory":[{

"device_ipaddress":<String_value>,
```

```
"device_type":<String_value>,

"act_id":<String_value>,

"agent_id":<String_value>,

"inventory_status":<String_value>,

"devices":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/inventory **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"inventory":[{

"device_ipaddress":<String_value>,

"device_type":<String_value>,

"act_id":<String_value>,
```

```
"agent_id":<String_value>,

"inventory_status":<String_value>,

"devices":<String_value>}]}
```

inventory_status

Configuration for Inventory Status resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
device_ip_address	<String>	Read-write	IP Address for device(XEN | ANY GUEST VPX). Minimum length = 1 Maximum length = 64
type	<String>	Read-write	Type of device, (Xen | Citrix ADC | Any VPX). Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key for all the task logs.
status_code	<Integer>	Read-only	Error status code if any..
status	<String>	Read-only	Status.
activity	<String>	Read-only	Activity.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
message	<String>	Read-only	Message.
act_id	<String>	Read-only	Activity Id.
is_last	<Boolean>	Read-only	is_last.
starttime	<Integer>	Read-only	Start Time.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/inventory_status **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/inventory_status **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of inventory_status resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/inventory_status **? pagesize=#no;pageno=#no** Use this query-parameter to get the inventory_status resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/inventory_status **?count=yes** Use this query-parameter to get the count of inventory_status resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"inventory_status":[{

"status_code":<Integer_value>,

"status":<String_value>,

"activity":<String_value>,

"device_ip_address":<String_value>,

"message":<String_value>,

"act_id":<String_value>,

"is_last":<Boolean_value>,

"starttime":<Integer_value>,

"type":<String_value>,

"id":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/inventory_status/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"inventory_status":[{

"status_code":<Integer_value>,

"status":<String_value>,

"activity":<String_value>,

"device_ip_address":<String_value>,

"message":<String_value>,

"act_id":<String_value>,

"is_last":<Boolean_value>,

"starttime":<Integer_value>,

"type":<String_value>,

"id":<String_value>}]}
```

managed_device

Configuration for Managed Device resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name .

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 256
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
description	<String>	Read-write	Description of managed device. Minimum length = 1 Maximum length = 512
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1 Maximum length = 64
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
is_managed	<Boolean>	Read-write	Is Managed.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device. Minimum length = 1 Maximum length = 128
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device. Minimum length = 1 Maximum length = 64
last_updated_time	<Double>	Read-write	Last Updated Time.
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.
id	<String>	Read-write	Id is system generated key for all the managed devices.
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
ipv6_address	<String>	Read-write	IPv6 Address.
license_edition	<String>	Read-write	Edition of instance.
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
device_family	<String>	Read-write	Device Family. Minimum length = 1 Maximum length = 64
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
type	<String>	Read-write	Type of device, (Xen | NS). Minimum length = 1 Maximum length = 64
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal network (not required to be set for SDWAN).
node_id	<String>	Read-write	Node identification of a device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1 Maximum length = 64
std_bw_total	<Integer>	Read-write	Standard Bandwidth.
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			A-B, and for Cluster it will be CLIP. Minimum length = 1 Maximum length = 128
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1 Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
instance_state	<String>	Read-only	State of device, UP only if device accessible.
reason	<String>	Read-only	Reason of failure for this managed device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
upsince	<String>	Read-only	Upsince of managed device.
tenant_id	<String>	Read-only	Tenant ID.
device_uuid	<String>	Read-only	Device UUID.
do_config	<Boolean>	Read-only	Do default config for managed device.
host_id	<String>	Read-only	Host ID.
version	<String>	Read-only	Device Version.
sysid	<String>	Read-only	System ID.
uptime	<String>	Read-only	Uptime of device.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.
status	<String>	Read-only	Status of managed device.
systemname	<String>	Read-only	Device System Name.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- ADD
- GET (ALL)
- DELETE
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ADD

URL: https://<MGMT-IP>/nitro/v2/config/managed_device?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"managed_device": {

<b>"ip_address":<String_value></b>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"profile_password":<String_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"last_updated_time":<Double_value>,

"gateway_ipv6":<String_value>,

"plt_bw_total":<Integer_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"name":<String_value>,

"description":<String_value>,
```

```
"ent_bw_available":<Integer_value>,

"type":<String_value>,

"mastools_version":<String_value>,

"gateway":<String_value>,

"ent_bw_total":<Integer_value>,

"config_type":<Integer_value>,

"vcpu_config":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"is_managed":<Boolean_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"instance_mode":<String_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"managed_device":[{

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"instance_state":<String_value>,

"reason":<String_value>,

"name":<String_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vcpu_config":<Integer_value>,

"tenant_id":<String_value>,
```

```
"device_uuid":<String_value>,

"netmask":<String_value>,

"do_config":<Boolean_value>,

"ent_bw_config":<Integer_value>,

"datacenter_id":<String_value>,

"host_id":<String_value>,

"version":<String_value>,

"is_managed":<Boolean_value>,

"instance_mode":<String_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"seq_no":<Double_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,

"last_updated_time":<Double_value>,

"plt_bw_total":<Integer_value>,

"uptime":<String_value>,

"id":<String_value>,

"mgmt_ip_address":<String_value>,

"ipv6_address":<String_value>,
```

```
"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"type":<String_value>,

"gateway":<String_value>,

"status":<String_value>,

"systemname":<String_value>,

"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"ping_state":<Integer_value>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"parent_inventory":<Boolean_value>,

"profile_password":<String_value>,

"act_id":<String_value>,

"profile_username":<String_value>,
```

```
"default_password":<String_value>,

"sync_operation":<Boolean_value>}]}
```

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/managed_device **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/managed_device **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of managed_device resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/managed_device **? pagesize=#no;pageno=#no** Use this query-parameter to get the managed_device resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/managed_device **?count=yes** Use this query-parameter to get the count of managed_device resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"managed_device":[{

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"instance_state":<String_value>,

"reason":<String_value>,

"name":<String_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vcpu_config":<Integer_value>,

"tenant_id":<String_value>,

"device_uuid":<String_value>,

"netmask":<String_value>,

"do_config":<Boolean_value>,

"ent_bw_config":<Integer_value>,
```

```
"datacenter_id":<String_value>,

"host_id":<String_value>,

"version":<String_value>,

"is_managed":<Boolean_value>,

"instance_mode":<String_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"seq_no":<Double_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,

"last_updated_time":<Double_value>,

"plt_bw_total":<Integer_value>,

"uptime":<String_value>,

"id":<String_value>,

"mgmt_ip_address":<String_value>,

"ipv6_address":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,
```

```
"license_grace_time":<Integer_value>,

"type":<String_value>,

"gateway":<String_value>,

"status":<String_value>,

"systemname":<String_value>,

"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"ping_state":<Integer_value>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"parent_inventory":<Boolean_value>,

"profile_password":<String_value>,

"act_id":<String_value>,

"profile_username":<String_value>,

"default_password":<String_value>,

"sync_operation":<Boolean_value>}]}
```

DELETE

URL: https://<MGMT-IP>/nitro/v2/config/managed_device/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

MODIFY

URL: https://<MGMT-IP>/nitro/v2/config/managed_device/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"managed_device":{

<b>"id":<String_value></b>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,
```

```
"profile_password":<String_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"last_updated_time":<Double_value>,

"gateway_ipv6":<String_value>,

"plt_bw_total":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"name":<String_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"type":<String_value>,

"mastools_version":<String_value>,

"gateway":<String_value>,

"ent_bw_total":<Integer_value>,

"config_type":<Integer_value>,

"vcpu_config":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,
```

```
"node_id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"is_managed":<Boolean_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"instance_mode":<String_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"managed_device":[{

"is_grace":<Boolean_value>,
```

```
"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"instance_state":<String_value>,

"reason":<String_value>,

"name":<String_value>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vcpu_config":<Integer_value>,

"tenant_id":<String_value>,

"device_uuid":<String_value>,

"netmask":<String_value>,

"do_config":<Boolean_value>,

"ent_bw_config":<Integer_value>,

"datacenter_id":<String_value>,

"host_id":<String_value>,

"version":<String_value>,
```

```
"is_managed":<Boolean_value>,

"instance_mode":<String_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"seq_no":<Double_value>,

"std_bw_available":<Integer_value>,

"internal_ip_address":<String_value>,

"sysid":<String_value>,

"last_updated_time":<Double_value>,

"plt_bw_total":<Integer_value>,

"uptime":<String_value>,

"id":<String_value>,

"mgmt_ip_address":<String_value>,

"ipv6_address":<String_value>,

"license_edition":<String_value>,

"cpu_license_type":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"type":<String_value>,

"gateway":<String_value>,
```

```
"status":<String_value>,

"systemname":<String_value>,

"config_type":<Integer_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"ping_state":<Integer_value>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"parent_inventory":<Boolean_value>,

"profile_password":<String_value>,

"act_id":<String_value>,

"profile_username":<String_value>,

"default_password":<String_value>,

"sync_operation":<Boolean_value>}]}
```

network_interface

Configuration for VM device network interface resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_mgmt_ifc	<Boolean>	Read-write	True if this is the management interface.
name_server	<String>	Read-write	Name Server. Minimum length = 1 Maximum length = 64
vrid_list_ipv6	<String>	Read-write	VRID List for Interface/Channel for IPV6 VMAC Generation.
gateway	<String>	Read-write	gateway. Minimum length = 1 Maximum length = 64
parent_id	<String>	Read-write	.
vrid_list_ipv4	<String>	Read-write	VRID List for Interface/Channel for IPV4 VMAC Generation.
is_member_ifc	<Boolean>	Read-write	True if this interface is member of a channel.
mac_address	<String>	Read-write	Mac Address. Minimum length = 1 Maximum length = 32
l2_enabled	<Boolean>	Read-write	L2 mode status of Interface.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	.
ip_address	<String>	Read-write	IP Address. Minimum length = 1 Maximum length = 64
netmask	<String>	Read-write	netmask. Minimum length = 1 Maximum length = 64
port_name	<String>	Read-write	Port name of the interface on the host machine. Minimum length = 1 Maximum length = 128
parent_name	<String>	Read-write	.
mac_mode	<String>	Read-write	Mac Mode, default for XenServer generated, generated for SVM generated, custom for User assigned. Minimum length = 1 Maximum length = 255
managed_device_id	<String>	Read-write	managed_device_id.
vlan	<Integer>	Read-write	This property is not supported.Use vlan_whitelist for vlan

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			configuration;ltbr;gtVLAN Id. Maximum value =
receiveuntagged	<Boolean>	Read-write	Receive Untagged Packets on Interface/Channel.
sdx_formation_network_id	<String>	Read-write	This property is deprecated;ltbr;gtSDX Formation Network Id of which this Interface is part of. Minimum length = 1 Maximum length = 128
vlan_whitelist	<String>	Read-write	VLAN Whitelist for Interface/Channel on VM Instance.
interface_name	<String>	Read-write	Name of this interface.
vrid_list_ipv4_array	<String[]>	Read-write	VRID List for Interface for IPV4 VMAC Generation.
vrid_list_ipv6_array	<String[]>	Read-write	VRID List for Interface for IPV6 VMAC Generation.
vlan_whitelist_array	<String[]>	Read-write	VLAN Whitelist for Interface on VM Instance.
is_vlan_applied	<Boolean>	Read-write	Is VLAN added on NetworkInterface of VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
device_channel_name	<String>	Read-only	LA Name on the actual VM.
xen_vf_index	<Integer>	Read-only	Index given by Xen when assigning free virtual function.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ping

Configuration for Ping resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
device_ipaddress	<String>	Read-write	Device IP Address. Minimum length = 1 Maximum length = 64
ping_status	<String>	Read-only	Status of ping.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ping **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ping":[{

"device_ipaddress":<String_value>,

"ping_status":<String_value>}]}
```

traceroute

Configuration for Traceroute resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
device_ipaddress	<String>	Read-write	Device IP Address. Minimum length = 1 Maximum length = 64
traceroute_status	<String>	Read-only	Status of traceroute.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/traceroute **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"traceroute":[{

"traceroute_status":<String_value>,

"device_ipaddress":<String_value>}]}
```

vm_device

Configuration for VM Device resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	Image Name, This parameter is used while provisioning VM Instance with XVA image, template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.
vm_memory_total	<Double>	Read-write	Total Memory of VM Instance in MB. 2048MB, 5120MB.
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
throughput_limit	<Double>	Read-write	Throughput Limit in Mbps set for VM Instance.
vlan_id_0_1	<Integer>	Read-write	VLAN id for the management interface 0/1. Maximum value =
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128
description	<String>	Read-write	Description of managed device. Minimum length = 1 Maximum length = 512
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
fips_partition_name	<String>	Read-write	FIPS Partition Name. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.
burst_priority	<Integer>	Read-write	Burst Priority of the VM Instance between 1 and 4.
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1 Maximum length = 64
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
is_managed	<Boolean>	Read-write	Is Managed.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device. Minimum length = 1 Maximum length = 128
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device. Minimum length = 1 Maximum length = 64
network_interfaces	<network_interface[]>	Read-write	Network Interfaces.
last_updated_time	<Double>	Read-write	Last Updated Time.
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.
host_ip_address	<String>	Read-write	Host IPAddress where VM is provisioned. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the managed devices.
ipv6_address	<String>	Read-write	IPv6 Address.
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
license_edition	<String>	Read-write	Edition of instance.
number_of_acu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
device_family	<String>	Read-write	Device Family. Minimum length = 1 Maximum length = 64
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..
la_mgmt	<Boolean>	Read-write	Bond consisting of management ports on VM Instance.
vlan_id_0_2	<Integer>	Read-write	VLAN id for the management interface 0/2. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
number_of_ssl_cores	<Integer>	Read-write	Assign number of ssl virtual functions to VM Instance.
type	<String>	Read-write	Type of device, (Xen | NS). Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	Assign throughput in Mbps to VM Instance.
template_name	<String>	Read-write	Template Name, This parameter is used while provisioning VM Instance with template, template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64
max_burst_throughput	<Double>	Read-write	Maximum burst throughput in Mbps of VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal network (not required to be set for SDWAN).
node_id	<String>	Read-write	Node identification of a device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1 Maximum length = 64
throughput_allocation_mode	<Integer>	Read-write	Throughput Allocation Mode: 0-Fixed, 1-Burst-able.
number_of_scu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
number_of_cores	<Integer>	Read-write	Number of cores that are assigned to VM Instance.
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be A-B, and for Cluster it will be CLIP. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
std_bw_total	<Integer>	Read-write	Standard Bandwidth.
domain_name	<String>	Read-write	Domain name of VM Device. Minimum length = 1 Maximum length = 128
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1 Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.
vlan_1_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/2 on VM Instance. Maximum value =
vrid_list_ipv4_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV4 VMAC Generation.
vlan_1_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 1/7 on VM Instance. Maximum value =
vrid_list_ipv6_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV6 VMAC Generation.
vrid_list_ipv4_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV4 VMAC Generation.
vrid_list_ipv6_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV6 VMAC Generation.
if_10_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/4 on VM Instance.
vlan_10_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/5 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV6 VMAC Generation.
vrid_list_ipv6_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV6 VMAC Generation.
if_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/5 on VM Instance.
receiveuntagged_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/4 on VM Instance.
vrid_list_ipv6_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV6 VMAC Generation.
vrid_list_ipv6_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 1/3for IPV6 VMAC Generation.
if_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/3 on VM Instance.
vrid_list_ipv6_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV6 VMAC Generation.
receiveuntagged_10 _4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/4 on VM Instance.
vrid_list_ipv6_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV6 VMAC Generation.
vlan_10_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/6 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/5 on VM Instance.
is_new_crypto	<Boolean>	Read-write	True if number_of_acu/ number_of_scu are used, false if number_of_ssl_core s is used.
vrid_list_ipv6_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV6 VMAC Generation.
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
if_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/7 on VM Instance.
receiveuntagged_10 _8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/8 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV6 VMAC Generation.
receiveuntagged_1_ 6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/6 on VM Instance.
vlan_1_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/4 on VM Instance. Maximum value =
if_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/2 on VM Instance.
receiveuntagged_1_ 5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/5 on VM Instance.
receiveuntagged_10 _6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Untagged Packets on 10/6 on VM Instance.
receiveuntagged_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/1 on VM Instance.
vrid_list_ipv4_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV4 VMAC Generation.
vrid_list_ipv6_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV6 VMAC Generation.
vrid_list_ipv6_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV6 VMAC Generation.
vrid_list_ipv4_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/2 on VM Instance.
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
vrid_list_ipv4_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV4 VMAC Generation.
receiveuntagged_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/7 on VM Instance.
vlan_10_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/4 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/7 on VM Instance.
receiveuntagged_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/1 on VM Instance.
vrid_list_ipv6_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV6 VMAC Generation.
if_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/1 on VM Instance.
if_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/8 on VM Instance.
l2_enabled	<Boolean>	Read-write	L2mode status of VM Instance.
vlan_10_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 10/7 on VM Instance. Maximum value =
vlan_10_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/1 on VM Instance. Maximum value =
receiveuntagged_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/2 on VM Instance.
reboot_vm_on_cpu_change	<Boolean>	Read-write	Reboot VMs on CPU change during resource allocation.
vlan_1_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/6 on VM Instance. Maximum value =
vrid_list_ipv4_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV4 VMAC Generation.
receiveuntagged_10_7	<Boolean>	Read-write	This property is deprecated by

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			network_interfaces;lt br;gtReceive Untagged Packets on 10/7 on VM Instance.
vrid_list_ipv4_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV4 VMAC Generation.
vlan_1_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/1 on VM Instance. Maximum value =
vrid_list_ipv6_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV6 VMAC Generation.
vlan_1_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/8 on VM Instance. Maximum value =
vlan_10_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 10/2 on VM Instance. Maximum value =
receiveuntagged_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/5 on VM Instance.
vrid_list_ipv4_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV4 VMAC Generation.
vrid_list_ipv4_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV4 VMAC Generation.
vrid_list_ipv4_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV4 VMAC Generation.
receiveuntagged_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			on 1/8 on VM Instance.
vrid_list_ipv4_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV4 VMAC Generation.
vlan_1_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/3 on VM Instance. Maximum value =
receiveuntagged_10 _3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/3 on VM Instance.
if_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/6 on VM Instance.
if_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/6 on VM Instance.
vlan_10_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVLAN for Network 10/3 on VM Instance. Maximum value =
vlan_1_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/5 on VM Instance. Maximum value =
if_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/3 on VM Instance.
vrid_list_ipv4_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV4 VMAC Generation.
if_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/4 on VM Instance.
vrid_list_ipv4_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/1 on VM Instance.
vrid_list_ipv4_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV4 VMAC Generation.
vlan_10_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/8 on VM Instance. Maximum value =
receiveuntagged_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/3 on VM Instance.
crypto_change_requires_reboot	<Boolean>	Read-write	True if the current changes made by user requires a reboot of the VM else false.
vrid_list_ipv4_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			IPV4 VMAC Generation.
if_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/8 on VM Instance.
vrid_list_ipv4_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV4 VMAC Generation.
vrid_list_ipv6_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for IPV6 VMAC Generation.
if_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/2 on VM Instance.
vrid_list_ipv6_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV6 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
sdxtools_http_port	<Integer>	Read-only	Http Port number to communicate to SDXTools.
instance_state	<String>	Read-only	State of device, UP only if device accessible.
reason	<String>	Read-only	Reason of failure for this managed device.
virtual_functions	<String>	Read-only	Virtual Functions assigned to VM Instance depending on SDX platform.
disk_space	<Double>	Read-only	Show Disk Space (GB) available to VM Instance.
upsince	<String>	Read-only	Upsince of managed device.
domid	<Double>	Read-only	DOM Id assigned to VM Instance viewed in hypervisor.
cpu_core_pe	<String>	Read-only	Packet Engine cores assigned to VM Instance.
tenant_id	<String>	Read-only	Tenant ID.
preferred_cpu_socket	<String>	Read-only	Preferred CPU Socket.
device_uuid	<String>	Read-only	Device UUID.
do_config	<Boolean>	Read-only	Do default config for managed device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
version	<String>	Read-only	Device Version.
host_id	<String>	Read-only	Host ID.
has_sdxtools	<Boolean>	Read-only	True if SDX Tools are installed on this VM.
vm_memory_used	<Double>	Read-only	Memory Used (MB) by VM Instance.
number_of_cpu	<Integer>	Read-only	Number of CPU that is assigned to VM Instance.
is_swg	<Boolean>	Read-only	Boolean to indicate whether a VM is SWG.
sysid	<String>	Read-only	System ID.
vm_state	<String>	Read-only	State of Virtual Machine (Running | Halted).
uuid	<String>	Read-only	UUID of VM Instance generated by Hypervisor.
cpu_core_mgmt	<String>	Read-only	Management CPU cores assigned to VM Instance.
ssl_virtual_functions	<String>	Read-only	SSL Virtual Functions assigned to VM Instance.
uptime	<String>	Read-only	Uptime of device.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
assigned_cpu_socket	<String>	Read-only	Assigned CPU Socket 0 or 1.
sdxtools_public_key	<String>	Read-only	SDXTools HTTPS Public Key.
status	<String>	Read-only	Status of managed device.
vm_description	<String>	Read-only	Description of the VM instance.
systemname	<String>	Read-only	Device System Name.
mac_eth0	<String>	Read-only	MAC Address of eth0 on VM Instance.
vm_rx	<Double>	Read-only	In Throughput of VM Instance in Mbps.
disk_allocation	<String>	Read-only	Disk allocation for VM Instance.
vm_tx	<Double>	Read-only	Out Throughput of VM Instance in Mbps.
vm_memory_free	<Double>	Read-only	Free Memory (MB) available in VM Instance.
vm_memory_usage	<Double>	Read-only	Memory Usage (%) of VM Instance.
sdxtools_version	<String>	Read-only	SDXTools version running on VM Instance.
vm_cpu_usage	<Double>	Read-only	CPU Usage (%) of VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/vm_device **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/vm_device **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of vm_device resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/vm_device **? pagesize=#no;pageno=#no** Use this query-parameter to get the vm_device resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/vm_device **?count=yes** Use this query-parameter to get the count of vm_device resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"vm_device":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,
```

```
"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,
```

```
"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,
```

```
"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,
```

```
"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,
```

```
"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,
```

```
"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,
```

```
"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,
```

```
"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,
```

```
"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,
```

```
"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,
```

```
"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/vm_device/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"vm_device":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,
```

```
"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,
```

```
"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,
```

```
"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,
```

```
"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,
```

```
"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,
```

```
"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,
```

```
"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,
```

```
"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,
```

```
"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,
```

```
"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,
```

```
"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

Licenses

- license_expiry_info
- license_file
- license_info
- license_proxy_server
- license_server
- license_server_pool
- license_user_detail
- sdx_license

license_expiry_info

Configuration for Citrix ADM License Expiry Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
expirycount	<Integer>	Read-write	Expiry Count.
feature	<String>	Read-write	Feature Name.
expirydays	<Integer>	Read-write	Expiry Days.
id	<String>	Read-write	Id is system generated key.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get

URL: https://<MGMT-IP>/nitro/v2/config/license_expiry_info **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_expiry_info":[{

"expirycount":<Integer_value>,

"feature":<String_value>,

"expirydays":<Integer_value>,

"id":<String_value>}]}
```

license_file

Configuration for License File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/license_file/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/license_file **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/license_file **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of license_file resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/license_file **? pagesize=#no;pageno=#no** Use this query-parameter to get the license_file resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/license_file **?count=yes** Use this query-parameter to get the count of license_file resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_file":[{

"file_location_path":<String_value>,
```

```
"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/license_file/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_file":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,
```

```
"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/license_file **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

license_info

Configuration for License Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
burst_license_available	<Integer>	Read-write	Burst license available for this license.
sync_operation	<Boolean>	Read-write	sync operation.
vpx_license_used	<Integer>	Read-write	Total license used by ADC/SDX.
id	<String>	Read-write	System generated value.
license_type	<String>	Read-only	License Type [Retail, NFR, Eval, TP, System].
user_info	<license_user_detail[]>	Read-only	userInfo.
feature	<String>	Read-only	Feature.
license_used	<Integer>	Read-only	Used Licenses.
act_id	<String>	Read-only	Activity Id.
product	<String>	Read-only	Product.
license_available	<Integer>	Read-only	Available Licenses.

Operations

(click to see Properties)

- APPLY
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

apply

URL: https://<MGMT-IP>/nitro/v2/config/license_info/id_value<String>?action=apply;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"license_info": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/license_info **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_info":[{

"license_type":<String_value>,

"user_info":[{
```

```
"no_of_license":<Integer_value>,

"parent_name":<String_value>,

"time":<Integer_value>,

"name":<String_value>,

"id":<String_value>,

"node":<String_value>,

"display":<String_value>,

"parent_id":<String_value>}],

"feature":<String_value>,

"burst_license_available":<Integer_value>,

"license_used":<Integer_value>,

"sync_operation":<Boolean_value>,

"act_id":<String_value>,

"vpx_license_used":<Integer_value>,

"id":<String_value>,

"product":<String_value>,

"license_available":<Integer_value>}]}
```

license_proxy_server

Configuration for License proxy configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
server	<String>	Read-write	IP address of License Web proxy server. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
port	<Integer>	Read-write	Port number of License Web proxy server.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/license_proxy_server?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"license_proxy_server": {

"id":<String_value>,

"server":<String_value>,
```

```
"port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_proxy_server":[{

"server":<String_value>,

"id":<String_value>,

"port":<Integer_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/license_proxy_server/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/license_proxy_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/license_proxy_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of license_proxy_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/license_proxy_server **? pagesize=#no;pageno=#no** Use this query-parameter to get the license_proxy_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/license_proxy_server **?count=yes** Use this query-parameter to get the count of license_proxy_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_proxy_server":[{

"server":<String_value>,

"id":<String_value>,
```

```
"port":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/license_proxy_server/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_proxy_server":[{

"server":<String_value>,

"id":<String_value>,

"port":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/license_proxy_server/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"license_proxy_server":{

"id":<String_value>,

"server":<String_value>,

"port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_proxy_server":[{

"server":<String_value>,

"id":<String_value>,

"port":<Integer_value>}]}
```

license_server

Configuration for License server information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
server	<String>	Read-write	Server/IP address of License server. Minimum length = 1 Maximum length = 255
id	<String>	Read-write	Id is system generated key.
port	<Integer>	Read-write	License port. Maximum value =
grace_time_left	<Integer>	Read-only	grace time remaining left in Hours.
forceupdateip	<Boolean>	Read-only	Change the license server ip and retain the capacity.
connection_status	<Boolean>	Read-only	License server is connected.
grace	<Boolean>	Read-only	Box is running on grace.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/license_server?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"license_server": {

<b>"server":<String_value></b>,

"id":<String_value>,

"port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_server":[{

"grace_time_left":<Integer_value>,

"forceupdateip":<Boolean_value>,

"server":<String_value>,

"id":<String_value>,

"connection_status":<Boolean_value>,

"grace":<Boolean_value>,

"port":<Integer_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/license_server/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/license_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/license_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of license_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/license_server **? pagesize=#no;pageno=#no** Use this query-parameter to get the license_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/license_server **?count=yes** Use this query-parameter to get the count of license_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_server":[{

"grace_time_left":<Integer_value>,

"forceupdateip":<Boolean_value>,

"server":<String_value>,

"id":<String_value>,

"connection_status":<Boolean_value>,

"grace":<Boolean_value>,

"port":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/license_server/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_server":[{

"grace_time_left":<Integer_value>,

"forceupdateip":<Boolean_value>,

"server":<String_value>,

"id":<String_value>,

"connection_status":<Boolean_value>,

"grace":<Boolean_value>,

"port":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/license_server/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"license_server":{

<b>"server":<String_value></b>,

"id":<String_value>,

"port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_server":[{

"grace_time_left":<Integer_value>,

"forceupdateip":<Boolean_value>,

"server":<String_value>,

"id":<String_value>,

"connection_status":<Boolean_value>,

"grace":<Boolean_value>,

"port":<Integer_value>}]}
```

license_server_pool

Configuration for License pool capacity of license server resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
std_bw_available	<Double>	Read-write	Available Standard Bandwidth pool in Gbps.
plt_bw_available	<Double>	Read-write	Available Platinum Bandwidth pool in Gbps.
std_bw_total	<Double>	Read-write	Total Standard Bandwidth pool in Gbps.
ent_bw_available	<Double>	Read-write	Available Enterprise Bandwidth pool in Gbps.
ent_bw_total	<Double>	Read-write	Total Enterprise Bandwidth pool in Gbps.
instance_total	<Double>	Read-write	Total instance pool.
plt_bw_total	<Double>	Read-write	Total Platinum Bandwidth pool in Gbps.
instance_available	<Double>	Read-write	Available instance pool.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/license_server_pool **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/license_server_pool **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of license_server_pool resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/license_server_pool **? pagesize=#no;pageno=#no** Use this query-parameter to get the license_server_pool resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/license_server_pool **?count=yes** Use this query-parameter to get the count of license_server_pool resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"license_server_pool":[{

"std_bw_available":<Double_value>,

"plt_bw_available":<Double_value>,

"std_bw_total":<Double_value>,

"ent_bw_available":<Double_value>,

"ent_bw_total":<Double_value>,

"instance_total":<Double_value>,

"plt_bw_total":<Double_value>,

"instance_available":<Double_value>}]}
```

license_user_detail

Configuration for License User Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
no_of_license	<Integer>	Read-write	No of licenses.
parent_name	<String>	Read-write	.
time	<Integer>	Read-write	Time.
name	<String>	Read-write	Name. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	.
node	<String>	Read-write	Node. Minimum length = 1 Maximum length = 128
display	<String>	Read-write	display. Minimum length = 1 Maximum length = 64
parent_id	<String>	Read-write	.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

sdx_license

Configuration for License Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
std_bw_available	<String>	Read-only	Available standard Bandwidth in Mbps.
max_number_of_ns_instances	<Integer>	Read-only	Maximum Citrix ADC Instances.
std_bw_config	<String>	Read-only	Configured Standard Bandwidth in Mbps.
ent_bw_total	<String>	Read-only	Total enterprise Bandwidth in Mbps.
act_id	<String>	Read-only	Activity Id.
available_throughput	<String>	Read-only	Available Throughput in Mbps.
instances_config	<Integer>	Read-only	Instances Configured .
plt_bw_total	<String>	Read-only	Maximum platinum Bandwidth in Mbps.
platform	<String>	Read-only	Platform.
max_number_of_instances	<Integer>	Read-only	Maximum Instances.
max_number_of_br_instances	<Integer>	Read-only	Maximum Citrix SD-WAN Instances.
plt_bw_available	<String>	Read-only	Available platinum Bandwidth in Mbps.
ent_bw_config	<String>	Read-only	Configured Enterpise Bandwidth in Mbps.
cluster	<Boolean>	Read-only	Cluster License.
available_number_of_ns_instances	<Integer>	Read-only	Available Citrix ADC Instances (Shared).

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
max_throughput	<String>	Read-only	Maximum Throughput in Mbps.
license_server_name	<String>	Read-only	License Server Configured in SDX .
std_bw_total	<String>	Read-only	Total standard Bandwidth in Mbps.
ent_bw_available	<String>	Read-only	Available enterprise Bandwidth in Mbps.
plt_bw_config	<String>	Read-only	Configured platinum Bandwidth in Mbps.
is_pooled_license	<Boolean>	Read-only	Platform license for Triscaler feature.
available_number_of_instances	<Integer>	Read-only	Available Instances.

Operations

(click to see Properties)

- APPLY
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

apply

URL: https://<MGMT-IP>/nitro/v2/config/sdx_license?action=apply;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sdx_license": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/sdx_license **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_license":[{

"std_bw_available":<String_value>,

"max_number_of_ns_instances":<Integer_value>,

"std_bw_config":<String_value>,

"ent_bw_total":<String_value>,
```

```
"act_id":<String_value>,

"available_throughput":<String_value>,

"instances_config":<Integer_value>,

"plt_bw_total":<String_value>,

"platform":<String_value>,

"max_number_of_instances":<Integer_value>,

"max_number_of_br_instances":<Integer_value>,

"plt_bw_available":<String_value>,

"ent_bw_config":<String_value>,

"cluster":<Boolean_value>,

"available_number_of_ns_instances":<Integer_value>,

"max_throughput":<String_value>,

"license_server_name":<String_value>,

"std_bw_total":<String_value>,

"ent_bw_available":<String_value>,

"sync_operation":<Boolean_value>,

"plt_bw_config":<String_value>,

"is_pooled_license":<Boolean_value>,

"available_number_of_instances":<Integer_value>}]}
```


Network-Functions

- Authentication

- Cache-Redirection
- Citrix-Gateway
- Common-Resources
- Content-Switching
- GSLB
- HAProxy
- Load-Balancing

Authentication

- ns_authenticationvserver

ns_authenticationvserver

Configuration for Citrix ADC Authvserver Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
user_managed	<Boolean>	Read-write	User Managed Idenfication.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100
export_list	<String>	Read-write	Export List present for the given LB VServer.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key.
vsvr_ip_address	<String>	Read-write	Vserver IP Address. Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	throughput value.
icalog	<String>	Read-write	ICA log.
appflow_policy_rule	<String>	Read-write	AppFlow policy rule present.
es4nslog	<String>	Read-write	ESNS enable.
transport_mode	<String>	Read-write	Transport type - IPFix or LogStream.
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
instance_license	<String>	Read-write	Feature License for Citrix ADC Instance, needs to be set while provisioning (standard, enterprise, platinum). Minimum length = 1 Maximum length = 64
agent_list	<String>	Read-write	Agent List, on which traffic will flow.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
export_option	<String>	Read-write	Export Options present for the given LB VServer.
appflowlog_effective	<String>	Read-write	Appflowlog effective (validating AppFlow config) enabled or disabled.
appflowlog	<String>	Read-write	Appflowlog enabled or disabled.
csvip_name	<String>	Read-write	CSVIP Name.
svc_id	<String>	Read-write	Service ID.
is_throughput_req	<Boolean>	Read-write	Throghput required.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
svc_name	<String>	Read-write	Service Name.
svc_grp_name	<String>	Read-write	Service Group Name.
svc_grp_id	<String>	Read-write	Service Group ID.
is_bot_policy_bound	<Boolean>	Read-write	Is Bot Policy bound to vserver..
csvip_id	<String>	Read-write	CSVIP ID.
app_security	<String>	Read-write	App Security to know whether bot and appfw policies are bound..
is_appfw_policy_bound	<Boolean>	Read-write	Is Appfw Policy bound to vserver..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
targetlbvserver	<String>	Read-only	Default Target LBVserver.
hits	<String>	Read-only	Traffic Domain.
license_type	<Integer>	Read-only	License Type : 0 - not licensed, 1 - manual, 2 - vpx, 3 - cpx, 4 - auto, 5 - cpx-sidecar/free.
state	<String>	Read-only	State of VServer.
cachetype	<String>	Read-only	Cache Type.
device_type	<String>	Read-only	Device Type : vpx, cpx, sdwan, blx, cpx-sidecar.
display_name	<String>	Read-only	Display Name of Vserver.
comment	<String>	Read-only	Vserver comment.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_authenticationvserver?action=enable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_authenticationvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,
```

```
"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_authenticationvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,
```

```
"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,
```

```
"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,
```

```
"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_authenticationvserver?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_authenticationvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,
```

```
"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_authenticationvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,
```

```
"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,
```

```
"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_authenticationvserver **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_authenticationvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,
```

```
"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,
```

```
"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_authenticationvserver?action=disable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_authenticationvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,
```

```
"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_authenticationvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,
```

```
"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,
```

```
"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,
```

```
"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

Cache-Redirection

- ns_crvserver

ns_crvserver

Configuration for Citrix ADC CRvserver Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
user_managed	<Boolean>	Read-write	User Managed Idenfication.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
export_list	<String>	Read-write	Export List present for the given LB VServer.
id	<String>	Read-write	Id is system generated key.
vsvr_ip_address	<String>	Read-write	Vserver IP Address. Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	throughput value.
icalog	<String>	Read-write	ICA log.
appflow_policy_rule	<String>	Read-write	AppFlow policy rule present.
es4nslog	<String>	Read-write	ESNS enable.
transport_mode	<String>	Read-write	Transport type - IPFix or LogStream.
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
instance_license	<String>	Read-write	Feature License for Citrix ADC Instance, needs to be set while provisioning (standard, enterprise, platinum). Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
agent_list	<String>	Read-write	Agent List, on which traffic will flow.
export_option	<String>	Read-write	Export Options present for the given LB VServer.
appflowlog_effective	<String>	Read-write	Appflowlog effective (validating AppFlow config) enabled or disabled.
appflowlog	<String>	Read-write	Appflowlog enabled or disabled.
csvip_name	<String>	Read-write	CSVIP Name.
svc_id	<String>	Read-write	Service ID.
is_throughput_req	<Boolean>	Read-write	Throghput required.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
svc_name	<String>	Read-write	Service Name.
svc_grp_name	<String>	Read-write	Service Group Name.
svc_grp_id	<String>	Read-write	Service Group ID.
is_bot_policy_bound	<Boolean>	Read-write	Is Bot Policy bound to vserver..
csvip_id	<String>	Read-write	CSVIP ID.
app_security	<String>	Read-write	App Security to know whether bot and

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			appfw policies are bound..
is_appfw_policy_bound	<Boolean>	Read-write	Is Appfw Policy bound to vserver..
targetlbvserver	<String>	Read-only	Default Target LBVserver.
hits	<String>	Read-only	Traffic Domain.
license_type	<Integer>	Read-only	License Type : 0 - not licensed, 1 - manual, 2 - vpx, 3 - cpx, 4 - auto, 5 - cpx-sidecar/free.
state	<String>	Read-only	State of VServer.
cachetype	<String>	Read-only	Cache Type.
device_type	<String>	Read-only	Device Type : vpx, cpx, sdwan, blx, cpx-sidecar.
display_name	<String>	Read-only	Display Name of Vserver.
comment	<String>	Read-only	Vserver comment.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_crvserver?action=enable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_crvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,
```

```
"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_crvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,
```

```
"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,
```

```
"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,
```

```
"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_crvserver?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_crvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,
```

```
"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_crvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,
```

```
"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,
```

```
"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_crvserver **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_crvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,
```

```
"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,
```

```
"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_crvserver?action=disable;onerror=<String_value>

HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_crvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,
```

```
"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_crvserver":[{

"user_managed":<Boolean_value>,
```

```
"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,
```

```
"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,
```

```
"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

Citrix-Gateway

- ns_vpnvserver

ns_vpnvserver

Configuration for Citrix ADC VPNvserver Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
user_managed	<Boolean>	Read-write	User Managed Idenfication.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100
export_list	<String>	Read-write	Export List present for the given LB VServer.
id	<String>	Read-write	Id is system generated key.
vsvr_ip_address	<String>	Read-write	Vserver IP Address. Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	throughput value.
icalog	<String>	Read-write	ICA log.
appflow_policy_rule	<String>	Read-write	AppFlow policy rule present.
es4nslog	<String>	Read-write	ESNS enable.
transport_mode	<String>	Read-write	Transport type - IPFix or LogStream.
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
instance_license	<String>	Read-write	Feature License for Citrix ADC Instance, needs to be set while provisioning (standard, enterprise, platinum). Minimum length = 1 Maximum length = 64
agent_list	<String>	Read-write	Agent List, on which traffic will flow.
export_option	<String>	Read-write	Export Options present for the given LB VServer.
appflowlog_effective	<String>	Read-write	Appflowlog effective (validating AppFlow config) enabled or disabled.
appflowlog	<String>	Read-write	Appflowlog enabled or disabled.
csvip_name	<String>	Read-write	CSVIP Name.
svc_id	<String>	Read-write	Service ID.
is_throughput_req	<Boolean>	Read-write	Throghput required.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
svc_name	<String>	Read-write	Service Name.
svc_grp_name	<String>	Read-write	Service Group Name.
svc_grp_id	<String>	Read-write	Service Group ID.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_bot_policy_bound	<Boolean>	Read-write	Is Bot Policy bound to vserver..
csvip_id	<String>	Read-write	CSVIP ID.
app_security	<String>	Read-write	App Security to know whether bot and appfw policies are bound..
is_appfw_policy_bound	<Boolean>	Read-write	Is Appfw Policy bound to vserver..
targetlbvserver	<String>	Read-only	Default Target LBVserver.
hits	<String>	Read-only	Traffic Domain.
license_type	<Integer>	Read-only	License Type : 0 - not licensed, 1 - manual, 2 - vpx, 3 - cpx, 4 - auto, 5 - cpx-sidecar/free.
state	<String>	Read-only	State of VServer.
cachetype	<String>	Read-only	Cache Type.
device_type	<String>	Read-only	Device Type : vpx, cpx, sdwan, blx, cpx-sidecar.
display_name	<String>	Read-only	Display Name of Vserver.
comment	<String>	Read-only	Vserver comment.

Operations

(click to see Properties)

- ENABLE

- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_vpnvserver?action=enable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_vpnvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,
```

```
"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vpnvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,
```

```
"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,
```

```
"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,
```

```
"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_vpnvserver?action=poll;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_vpnvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,
```

```
"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,
```

```
"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vpnvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,
```

```
"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,
```

```
"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_vpnvserver **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vpnvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,
```

```
"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,
```

```
"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_vpnvserver?action=disable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_vpnvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,
```

```
"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vpnvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,
```

```
"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,
```

```
"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

Common-Resources

- ns_sslvserver

ns_sslvserver

Configuration for Citrix ADC SSLvserver Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
denysslreneg	<String>	Read-write	denysslreneg.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
sessionticketkeyrefresh	<String>	Read-write	sessionticketkeyrefresh.
quantumsize	<Integer>	Read-write	quantumsize.
cipherurl	<String>	Read-write	cipherurl.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
ssltriggertimeout	<Integer>	Read-write	ssltriggertimeout.
pushflag	<Integer>	Read-write	pushflag.
allowextendedmastersecret	<String>	Read-write	allowextendedmastersecret.
commonname	<String>	Read-write	commonname.
ssllogprofile	<String>	Read-write	ssllogprofile.
name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100
dhcount	<Integer>	Read-write	dhcount.
tls13sessionticketsperauthcontext	<Integer>	Read-write	tls13sessionticketsperauthcontext.
pushenctriggertimeout	<Integer>	Read-write	pushenctriggertimeout.
dh	<Boolean>	Read-write	dh.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hsts	<Boolean>	Read-write	hsts.
serverauth	<String>	Read-write	serverauth.
tls13	<Boolean>	Read-write	tls13.
snienable	<String>	Read-write	snienable.
sslinterception	<String>	Read-write	sslinterception.
cipherredirect	<String>	Read-write	cipherredirect.
strictsigdigestcheck	<Boolean>	Read-write	strictsigdigestcheck.
sslimaxsessperserver	<Integer>	Read-write	sslimaxsessperserver.
clientcert	<String>	Read-write	clientcert.
prevsessionkeylifetime	<Integer>	Read-write	prevsessionkeylifetime.
sessionticketlifetime	<Integer>	Read-write	sessionticketlifetime.
dhfile	<String>	Read-write	dhfile.
managed	<Boolean>	Read-write	Managed vserver.
encrypttriggerpktcount	<Integer>	Read-write	encrypttriggerpktcount.
ssl2	<Boolean>	Read-write	ssl2.
sslredirect	<String>	Read-write	sslredirect.
insertionencoding	<String>	Read-write	insertionencoding.
ersa	<Boolean>	Read-write	ersa.
strictcachecks	<String>	Read-write	strictcachecks.
sslireneg	<String>	Read-write	sslireneg.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
redirectportrewrite	<String>	Read-write	redirectportrewrite.
sessreuse	<String>	Read-write	sessreuse.
ssl3	<Boolean>	Read-write	ssl3.
sessionticketkeydata	<String>	Read-write	sessionticketkeydata .
sessionticket	<String>	Read-write	sessionticket.
id	<String>	Read-write	Id is system generated key.
cleartextport	<String>	Read-write	cleartextport.
alpnprotocol	<String>	Read-write	alpnprotocol.
clientauthuseboundcachain	<String>	Read-write	clientauthuseboundcachain.
dhekeyexchangewithpsk	<String>	Read-write	dhekeyexchangewithpsk.
ocspstapling	<String>	Read-write	ocspstapling.
ersacount	<Integer>	Read-write	ersacount.
snihttphostmatch	<String>	Read-write	snihttphostmatch.
preload	<String>	Read-write	preload.
sesstimeout	<Integer>	Read-write	sesstimeout.
sendclosenotify	<String>	Read-write	sendclosenotify.
tls11	<Boolean>	Read-write	tls11.
skipclientcertpolicycheck	<String>	Read-write	skipclientcertpolicycheck.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
dropreqwithnohostheader	<String>	Read-write	dropreqwithnohostheader.
maxage	<Integer>	Read-write	maxage.
sessionkeylifetime	<Integer>	Read-write	sessionkeylifetime.
tls12	<Boolean>	Read-write	tls12.
includesubdomains	<Boolean>	Read-write	includesubdomains.
tls10	<Boolean>	Read-write	tls10.
dhkeyexpsizelimit	<String>	Read-write	dhkeyexpsizelimit.
ssliocspcheck	<String>	Read-write	ssliocspcheck.
pushenctrigger	<String>	Read-write	pushenctrigger.
clientauth	<String>	Read-write	clientauth.
zerorttearlydata	<String>	Read-write	zerorttearlydata.
sslprofile	<String>	Read-only	SSL Profile Name.
poll_time	<Integer>	Read-only	Last Polling Time.
display_name	<String>	Read-only	Display Name.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_sslvserver?action=enable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_sslvserver": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"commonname":<String_value>,

"name":<String_value>,

"dh":<Boolean_value>,

"tls13":<Boolean_value>,

"cipherredirect":<String_value>,

"snienable":<String_value>,

"strictsigdigestcheck":<Boolean_value>,

"sslimaxsessperserver":<Integer_value>,

"prevsessionkeylifetime":<Integer_value>,

"sessionticketlifetime":<Integer_value>,

"dhfile":<String_value>,

"ssl2":<Boolean_value>,

"encrypttriggerpktcount":<Integer_value>,

"sslredirect":<String_value>,

"ersa":<Boolean_value>,

"sslireneg":<String_value>,
```

```
"redirectportrewrite":<String_value>,

"sessionticketkeydata":<String_value>,

"sessionticket":<String_value>,

"dhekeyexchangewithpsk":<String_value>,

"snihttphostmatch":<String_value>,

"preload":<String_value>,

"sendclosenotify":<String_value>,

"tls11":<Boolean_value>,

"maxage":<Integer_value>,

"tls10":<Boolean_value>,

"ssliocspcheck":<String_value>,

"pushenctrigger":<String_value>,

"denysslreneg":<String_value>,

"sessionticketkeyrefresh":<String_value>,

"quantumsize":<Integer_value>,

"cipherurl":<String_value>,

"ssltriggertimeout":<Integer_value>,

"pushflag":<Integer_value>,

"ssllogprofile":<String_value>,

"allowextendedmastersecret":<String_value>,

"tls13sessionticketsperauthcontext":<Integer_value>,

"dhcount":<Integer_value>,
```

```
"pushenctriggertimeout":<Integer_value>,

"hsts":<Boolean_value>,

"serverauth":<String_value>,

"sslinterception":<String_value>,

"clientcert":<String_value>,

"managed":<Boolean_value>,

"insertionencoding":<String_value>,

"strictcachechecks":<String_value>,

"sessreuse":<String_value>,

"ssl3":<Boolean_value>,

"cleartextport":<String_value>,

"clientauthuseboundcachain":<String_value>,

"alpnprotocol":<String_value>,

"ocspstapling":<String_value>,

"ersacount":<Integer_value>,

"sesstimeout":<Integer_value>,

"skipclientcertpolicycheck":<String_value>,

"dropreqwithnohostheader":<String_value>,

"sessionkeylifetime":<Integer_value>,

"tls12":<Boolean_value>,

"includesubdomains":<Boolean_value>,

"dhkeyexpsizelimit":<String_value>,
```

```
"clientauth":<String_value>,

"zerorttearlydata":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_sslvserver":[{

"denysslreneg":<String_value>,

"sessionticketkeyrefresh":<String_value>,

"quantumsize":<Integer_value>,

"cipherurl":<String_value>,

"ns_ip_address":<String_value>,

"ssltriggertimeout":<Integer_value>,

"vsvr_type":<String_value>,

"pushflag":<Integer_value>,

"allowextendedmastersecret":<String_value>,

"commonname":<String_value>,

"ssllogprofile":<String_value>,

"name":<String_value>,

"dhcount":<Integer_value>,

"tls13sessionticketsperauthcontext":<Integer_value>,
```

```
"pushenctriggertimeout":<Integer_value>,

"dh":<Boolean_value>,

"hsts":<Boolean_value>,

"sslprofile":<String_value>,

"serverauth":<String_value>,

"tls13":<Boolean_value>,

"snienable":<String_value>,

"sslinterception":<String_value>,

"cipherredirect":<String_value>,

"strictsigdigestcheck":<Boolean_value>,

"sslimaxsessperserver":<Integer_value>,

"clientcert":<String_value>,

"vsvr_port":<Integer_value>,

"prevsessionkeylifetime":<Integer_value>,

"sessionticketlifetime":<Integer_value>,

"dhfile":<String_value>,

"managed":<Boolean_value>,

"encrypttriggerpktcount":<Integer_value>,

"ssl2":<Boolean_value>,

"sslredirect":<String_value>,

"insertionencoding":<String_value>,

"ersa":<Boolean_value>,
```

```
"strictcachecks":<String_value>,

"sslireneg":<String_value>,

"state":<String_value>,

"redirectportrewrite":<String_value>,

"sessreuse":<String_value>,

"ssl3":<Boolean_value>,

"sessionticketkeydata":<String_value>,

"sessionticket":<String_value>,

"id":<String_value>,

"cleartextport":<String_value>,

"alpnprotocol":<String_value>,

"clientauthuseboundcachain":<String_value>,

"dhekeyexchangewithpsk":<String_value>,

"poll_time":<Integer_value>,

"ocspstapling":<String_value>,

"ersacount":<Integer_value>,

"snihttphostmatch":<String_value>,

"vsvr_ip_address":<String_value>,

"preload":<String_value>,

"sesstimeout":<Integer_value>,

"sendclosenotify":<String_value>,

"tls11":<Boolean_value>,
```

```
"skipclientcertpolicycheck":<String_value>,

"dropreqwithnohostheader":<String_value>,

"maxage":<Integer_value>,

"sessionkeylifetime":<Integer_value>,

"tls12":<Boolean_value>,

"includesubdomains":<Boolean_value>,

"tls10":<Boolean_value>,

"dhkeyexpsizelimit":<String_value>,

"ssliocspcheck":<String_value>,

"pushenctrigger":<String_value>,

"clientauth":<String_value>,

"display_name":<String_value>,

"zerorttearlydata":<String_value>,

"partition_name":<String_value>,

"protocol_type":<String_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_sslvserver?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_sslvserver": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"commonname":<String_value>,

"name":<String_value>,

"dh":<Boolean_value>,

"tls13":<Boolean_value>,

"cipherredirect":<String_value>,

"snienable":<String_value>,

"strictsigdigestcheck":<Boolean_value>,

"sslimaxsessperserver":<Integer_value>,

"prevsessionkeylifetime":<Integer_value>,

"sessionticketlifetime":<Integer_value>,

"dhfile":<String_value>,

"ssl2":<Boolean_value>,

"encrypttriggerpktcount":<Integer_value>,

"sslredirect":<String_value>,

"ersa":<Boolean_value>,

"sslireneg":<String_value>,

"redirectportrewrite":<String_value>,
```

```
"sessionticketkeydata":<String_value>,

"sessionticket":<String_value>,

"dhekeyexchangewithpsk":<String_value>,

"snihttphostmatch":<String_value>,

"preload":<String_value>,

"sendclosenotify":<String_value>,

"tls11":<Boolean_value>,

"maxage":<Integer_value>,

"tls10":<Boolean_value>,

"ssliocspcheck":<String_value>,

"pushenctrigger":<String_value>,

"denysslreneg":<String_value>,

"sessionticketkeyrefresh":<String_value>,

"quantumsize":<Integer_value>,

"cipherurl":<String_value>,

"ssltriggertimeout":<Integer_value>,

"pushflag":<Integer_value>,

"ssllogprofile":<String_value>,

"allowextendedmastersecret":<String_value>,

"tls13sessionticketsperauthcontext":<Integer_value>,

"dhcount":<Integer_value>,

"pushenctriggertimeout":<Integer_value>,
```

```
"hsts":<Boolean_value>,

"serverauth":<String_value>,

"sslinterception":<String_value>,

"clientcert":<String_value>,

"managed":<Boolean_value>,

"insertionencoding":<String_value>,

"strictcachecks":<String_value>,

"sessreuse":<String_value>,

"ssl3":<Boolean_value>,

"cleartextport":<String_value>,

"clientauthuseboundcachain":<String_value>,

"alpnprotocol":<String_value>,

"ocspstapling":<String_value>,

"ersacount":<Integer_value>,

"sesstimeout":<Integer_value>,

"skipclientcertpolicycheck":<String_value>,

"dropreqwithnohostheader":<String_value>,

"sessionkeylifetime":<Integer_value>,

"tls12":<Boolean_value>,

"includesubdomains":<Boolean_value>,

"dhkeyexpsizelimit":<String_value>,

"clientauth":<String_value>,
```

```
"zerorttearlydata":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_sslvserver":[{

"denysslreneg":<String_value>,

"sessionticketkeyrefresh":<String_value>,

"quantumsize":<Integer_value>,

"cipherurl":<String_value>,

"ns_ip_address":<String_value>,

"ssltriggertimeout":<Integer_value>,

"vsvr_type":<String_value>,

"pushflag":<Integer_value>,

"allowextendedmastersecret":<String_value>,

"commonname":<String_value>,

"ssllogprofile":<String_value>,

"name":<String_value>,

"dhcount":<Integer_value>,

"tls13sessionticketsperauthcontext":<Integer_value>,

"pushenctriggertimeout":<Integer_value>,
```

```
"dh":<Boolean_value>,

"hsts":<Boolean_value>,

"sslprofile":<String_value>,

"serverauth":<String_value>,

"tls13":<Boolean_value>,

"snienable":<String_value>,

"sslinterception":<String_value>,

"cipherredirect":<String_value>,

"strictsigdigestcheck":<Boolean_value>,

"sslimaxsessperserver":<Integer_value>,

"clientcert":<String_value>,

"vsvr_port":<Integer_value>,

"prevsessionkeylifetime":<Integer_value>,

"sessionticketlifetime":<Integer_value>,

"dhfile":<String_value>,

"managed":<Boolean_value>,

"encrypttriggerpktcount":<Integer_value>,

"ssl2":<Boolean_value>,

"sslredirect":<String_value>,

"insertionencoding":<String_value>,

"ersa":<Boolean_value>,

"strictcachecks":<String_value>,
```

```
"sslireneg":<String_value>,

"state":<String_value>,

"redirectportrewrite":<String_value>,

"sessreuse":<String_value>,

"ssl3":<Boolean_value>,

"sessionticketkeydata":<String_value>,

"sessionticket":<String_value>,

"id":<String_value>,

"cleartextport":<String_value>,

"alpnprotocol":<String_value>,

"clientauthuseboundcachain":<String_value>,

"dhekeyexchangewithpsk":<String_value>,

"poll_time":<Integer_value>,

"ocspstapling":<String_value>,

"ersacount":<Integer_value>,

"snihttphostmatch":<String_value>,

"vsvr_ip_address":<String_value>,

"preload":<String_value>,

"sesstimeout":<Integer_value>,

"sendclosenotify":<String_value>,

"tls11":<Boolean_value>,

"skipclientcertpolicycheck":<String_value>,
```

```
"dropreqwithnohostheader":<String_value>,

"maxage":<Integer_value>,

"sessionkeylifetime":<Integer_value>,

"tls12":<Boolean_value>,

"includesubdomains":<Boolean_value>,

"tls10":<Boolean_value>,

"dhkeyexpsizelimit":<String_value>,

"ssliocspcheck":<String_value>,

"pushenctrigger":<String_value>,

"clientauth":<String_value>,

"display_name":<String_value>,

"zerorttearlydata":<String_value>,

"partition_name":<String_value>,

"protocol_type":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_sslvserver **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_sslvserver":[{

"denysslreneg":<String_value>,

"sessionticketkeyrefresh":<String_value>,

"quantumsize":<Integer_value>,

"cipherurl":<String_value>,

"ns_ip_address":<String_value>,

"ssltriggertimeout":<Integer_value>,

"vsvr_type":<String_value>,

"pushflag":<Integer_value>,

"allowextendedmastersecret":<String_value>,

"commonname":<String_value>,

"ssllogprofile":<String_value>,

"name":<String_value>,

"dhcount":<Integer_value>,

"tls13sessionticketsperauthcontext":<Integer_value>,

"pushenctriggertimeout":<Integer_value>,
```

```
"dh":<Boolean_value>,

"hsts":<Boolean_value>,

"sslprofile":<String_value>,

"serverauth":<String_value>,

"tls13":<Boolean_value>,

"snienable":<String_value>,

"sslinterception":<String_value>,

"cipherredirect":<String_value>,

"strictsigdigestcheck":<Boolean_value>,

"sslimaxsessperserver":<Integer_value>,

"clientcert":<String_value>,

"vsvr_port":<Integer_value>,

"prevsessionkeylifetime":<Integer_value>,

"sessionticketlifetime":<Integer_value>,

"dhfile":<String_value>,

"managed":<Boolean_value>,

"encrypttriggerpktcount":<Integer_value>,

"ssl2":<Boolean_value>,

"sslredirect":<String_value>,

"insertionencoding":<String_value>,

"ersa":<Boolean_value>,

"strictcachecks":<String_value>,
```

```
"sslireneg":<String_value>,

"state":<String_value>,

"redirectportrewrite":<String_value>,

"sessreuse":<String_value>,

"ssl3":<Boolean_value>,

"sessionticketkeydata":<String_value>,

"sessionticket":<String_value>,

"id":<String_value>,

"cleartextport":<String_value>,

"alpnprotocol":<String_value>,

"clientauthuseboundcachain":<String_value>,

"dhekeyexchangewithpsk":<String_value>,

"poll_time":<Integer_value>,

"ocspstapling":<String_value>,

"ersacount":<Integer_value>,

"snihttphostmatch":<String_value>,

"vsvr_ip_address":<String_value>,

"preload":<String_value>,

"sesstimeout":<Integer_value>,

"sendclosenotify":<String_value>,

"tls11":<Boolean_value>,

"skipclientcertpolicycheck":<String_value>,
```

```
"dropreqwithnohostheader":<String_value>,

"maxage":<Integer_value>,

"sessionkeylifetime":<Integer_value>,

"tls12":<Boolean_value>,

"includesubdomains":<Boolean_value>,

"tls10":<Boolean_value>,

"dhkeyexpsizelimit":<String_value>,

"ssliocspcheck":<String_value>,

"pushenctrigger":<String_value>,

"clientauth":<String_value>,

"display_name":<String_value>,

"zerorttearlydata":<String_value>,

"partition_name":<String_value>,

"protocol_type":<String_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_sslvserver?action=disable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_sslvserver": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"commonname":<String_value>,

"name":<String_value>,

"dh":<Boolean_value>,

"tls13":<Boolean_value>,

"cipherredirect":<String_value>,

"snienable":<String_value>,

"strictsigdigestcheck":<Boolean_value>,

"sslimaxsessperserver":<Integer_value>,

"prevsessionkeylifetime":<Integer_value>,

"sessionticketlifetime":<Integer_value>,

"dhfile":<String_value>,

"ssl2":<Boolean_value>,

"encrypttriggerpktcount":<Integer_value>,

"sslredirect":<String_value>,

"ersa":<Boolean_value>,

"sslireneg":<String_value>,

"redirectportrewrite":<String_value>,

"sessionticketkeydata":<String_value>,
```

```
"sessionticket":<String_value>,

"dhekeyexchangewithpsk":<String_value>,

"snihttphostmatch":<String_value>,

"preload":<String_value>,

"sendclosenotify":<String_value>,

"tls11":<Boolean_value>,

"maxage":<Integer_value>,

"tls10":<Boolean_value>,

"ssliocspcheck":<String_value>,

"pushenctrigger":<String_value>,

"denysslreneg":<String_value>,

"sessionticketkeyrefresh":<String_value>,

"quantumsize":<Integer_value>,

"cipherurl":<String_value>,

"ssltriggertimeout":<Integer_value>,

"pushflag":<Integer_value>,

"ssllogprofile":<String_value>,

"allowextendedmastersecret":<String_value>,

"tls13sessionticketsperauthcontext":<Integer_value>,

"dhcount":<Integer_value>,

"pushenctriggertimeout":<Integer_value>,

"hsts":<Boolean_value>,
```

```
"serverauth":<String_value>,

"sslinterception":<String_value>,

"clientcert":<String_value>,

"managed":<Boolean_value>,

"insertionencoding":<String_value>,

"strictcachecks":<String_value>,

"sessreuse":<String_value>,

"ssl3":<Boolean_value>,

"cleartextport":<String_value>,

"clientauthuseboundcachain":<String_value>,

"alpnprotocol":<String_value>,

"ocspstapling":<String_value>,

"ersacount":<Integer_value>,

"sesstimeout":<Integer_value>,

"skipclientcertpolicycheck":<String_value>,

"dropreqwithnohostheader":<String_value>,

"sessionkeylifetime":<Integer_value>,

"tls12":<Boolean_value>,

"includesubdomains":<Boolean_value>,

"dhkeyexpsizelimit":<String_value>,

"clientauth":<String_value>,

"zerorttearlydata":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_sslvserver":[{

"denysslreneg":<String_value>,

"sessionticketkeyrefresh":<String_value>,

"quantumsize":<Integer_value>,

"cipherurl":<String_value>,

"ns_ip_address":<String_value>,

"ssltriggertimeout":<Integer_value>,

"vsvr_type":<String_value>,

"pushflag":<Integer_value>,

"allowextendedmastersecret":<String_value>,

"commonname":<String_value>,

"ssllogprofile":<String_value>,

"name":<String_value>,

"dhcount":<Integer_value>,

"tls13sessionticketsperauthcontext":<Integer_value>,

"pushenctriggertimeout":<Integer_value>,

"dh":<Boolean_value>,
```

```
"hsts":<Boolean_value>,

"sslprofile":<String_value>,

"serverauth":<String_value>,

"tls13":<Boolean_value>,

"snienable":<String_value>,

"sslinterception":<String_value>,

"cipherredirect":<String_value>,

"strictsigdigestcheck":<Boolean_value>,

"sslimaxsessperserver":<Integer_value>,

"clientcert":<String_value>,

"vsvr_port":<Integer_value>,

"prevsessionkeylifetime":<Integer_value>,

"sessionticketlifetime":<Integer_value>,

"dhfile":<String_value>,

"managed":<Boolean_value>,

"encrypttriggerpktcount":<Integer_value>,

"ssl2":<Boolean_value>,

"sslredirect":<String_value>,

"insertionencoding":<String_value>,

"ersa":<Boolean_value>,

"strictcachecks":<String_value>,

"sslireneg":<String_value>,
```

```
"state":<String_value>,

"redirectportrewrite":<String_value>,

"sessreuse":<String_value>,

"ssl3":<Boolean_value>,

"sessionticketkeydata":<String_value>,

"sessionticket":<String_value>,

"id":<String_value>,

"cleartextport":<String_value>,

"alpnprotocol":<String_value>,

"clientauthuseboundcachain":<String_value>,

"dhekeyexchangewithpsk":<String_value>,

"poll_time":<Integer_value>,

"ocspstapling":<String_value>,

"ersacount":<Integer_value>,

"snihttphostmatch":<String_value>,

"vsvr_ip_address":<String_value>,

"preload":<String_value>,

"sesstimeout":<Integer_value>,

"sendclosenotify":<String_value>,

"tls11":<Boolean_value>,

"skipclientcertpolicycheck":<String_value>,

"dropreqwithnohostheader":<String_value>,
```

```
"maxage":<Integer_value>,

"sessionkeylifetime":<Integer_value>,

"tls12":<Boolean_value>,

"includesubdomains":<Boolean_value>,

"tls10":<Boolean_value>,

"dhkeyexpsizelimit":<String_value>,

"ssliocspcheck":<String_value>,

"pushenctrigger":<String_value>,

"clientauth":<String_value>,

"display_name":<String_value>,

"zerorttearlydata":<String_value>,

"partition_name":<String_value>,

"protocol_type":<String_value>}]}
```

Content-Switching

- ns_csvserver
- ns_csvserver_lbvserver_binding

ns_csvserver

Configuration for Citrix ADC CSvserver Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
user_managed	<Boolean>	Read-write	User Managed Idenfication.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100
export_list	<String>	Read-write	Export List present for the given LB VServer.
id	<String>	Read-write	Id is system generated key.
vsvr_ip_address	<String>	Read-write	Vserver IP Address. Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	throughput value.
icalog	<String>	Read-write	ICA log.
appflow_policy_rule	<String>	Read-write	AppFlow policy rule present.
es4nslog	<String>	Read-write	ESNS enable.
transport_mode	<String>	Read-write	Transport type - IPFix or LogStream.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
instance_license	<String>	Read-write	Feature License for Citrix ADC Instance, needs to be set while provisioning (standard, enterprise, platinum). Minimum length = 1 Maximum length = 64
agent_list	<String>	Read-write	Agent List, on which traffic will flow.
export_option	<String>	Read-write	Export Options present for the given LB VServer.
appflowlog_effective	<String>	Read-write	Appflowlog effective (validating AppFlow config) enabled or disabled.
appflowlog	<String>	Read-write	Appflowlog enabled or disabled.
csvip_name	<String>	Read-write	CSVIP Name.
svc_id	<String>	Read-write	Service ID.
is_throughput_req	<Boolean>	Read-write	Throghput required.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
svc_name	<String>	Read-write	Service Name.
svc_grp_name	<String>	Read-write	Service Group Name.
svc_grp_id	<String>	Read-write	Service Group ID.
is_bot_policy_bound	<Boolean>	Read-write	Is Bot Policy bound to vserver..
csvip_id	<String>	Read-write	CSVIP ID.
app_security	<String>	Read-write	App Security to know whether bot and appfw policies are bound..
is_appfw_policy_bound	<Boolean>	Read-write	Is Appfw Policy bound to vserver..
targetlbvserver	<String>	Read-only	Default Target LBVserver.
hits	<String>	Read-only	Traffic Domain.
license_type	<Integer>	Read-only	License Type : 0 - not licensed, 1 - manual, 2 - vpx, 3 - cpx, 4 - auto, 5 - cpx-sidecar/free.
state	<String>	Read-only	State of VServer.
cachetype	<String>	Read-only	Cache Type.
device_type	<String>	Read-only	Device Type : vpx, cpx, sdwan, blx, cpx-sidecar.
display_name	<String>	Read-only	Display Name of Vserver.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
comment	<String>	Read-only	Vserver comment.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_csvserver?action=enable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_csvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,
```

```
"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_csvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,
```

```
"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,
```

```
"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_csvserver?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_csvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,
```

```
"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_csvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,
```

```
"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,
```

```
"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,
```

```
"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_csvserver **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_csvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,
```

```
"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,
```

```
"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,
```

```
"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_csvserver?action=disable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_csvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,
```

```
"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_csvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,
```

```
"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,
```

```
"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

ns_csvserver_lbvserver_binding

Configuration for Citrix ADC CSvserver Binding Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
policyname	<String>	Read-write	CS Policy Name. Minimum length = 1 Maximum length = 64
csvsvr_name	<String>	Read-write	CS Vserver Name.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
priority	<String>	Read-only	priority.
hostname	<String>	Read-only	Hostname of the managed device.
display_name	<String>	Read-only	Display Name.
partition_name	<String>	Read-only	Partition Name.
targetlbvserver	<String>	Read-only	Target LB VServer.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_csvserver_lbvserver_binding **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_csvserver_lbvserver_binding":[{

"priority":<String_value>,

"policyname":<String_value>,

"csvsvr_name":<String_value>,

"hostname":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"targetlbvserver":<String_value>,

"id":<String_value>}]}
```

GSLB

- ns_gslbservice
- ns_gslbservicegroup
- ns_gslbservicegroupmember_binding
- ns_gslbvserver
- ns_gslbvserver_domain

ns_gslbservice

Configuration for Citrix ADC GSLB Services Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
svc_ip_address	<String>	Read-write	Service IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
svc_port	<Integer>	Read-write	Service Port. Minimum value = 1 Maximum value =
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
svc_name	<String>	Read-write	Service Name. Minimum length = 1 Maximum length = 100
publicip	<String>	Read-write	Public IP Address. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
appflowlog	<String>	Read-write	AppFlow is enabled at service of LBVServer.
vsvr_name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100
delay	<Double>	Read-write	Time, in seconds, allocated to the Citrix ADC appliance for a graceful shutdown of the service.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
svr_id	<String>	Read-write	Server ID.
lb_id	<String>	Read-write	LB ID.
graceful	<Boolean>	Read-write	Graceful shutdown of Service..
site_name	<String>	Read-only	Site Name.
svr_name	<String>	Read-only	Server Name associated with service.
svc_type	<String>	Read-only	Service Protocol (HTTP|FTP).
poll_time	<Integer>	Read-only	Last Polling Time.
svc_state	<String>	Read-only	Service State.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
display_name	<String>	Read-only	Display Name.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservice?action=enable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbservice": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_port":<Integer_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,
```

```
"svc_name":<String_value>,

"publicip":<String_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservice":[{

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"site_name":<String_value>,

"svr_name":<String_value>,

"id":<String_value>,

"svc_port":<Integer_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svc_type":<String_value>,

"poll_time":<Integer_value>,
```

```
"svc_state_timestamp":<Double_value>,

"type_svc":<String_value>,

"appflowlog":<String_value>,

"svc_state":<String_value>,

"display_name":<String_value>,

"partition_name":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"save_config":<Boolean_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"graceful":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservice?action=poll;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbservice": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_port":<Integer_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservice":[{

"ns_ip_address":<String_value>,
```

```
"svc_ip_address":<String_value>,

"site_name":<String_value>,

"svr_name":<String_value>,

"id":<String_value>,

"svc_port":<Integer_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svc_type":<String_value>,

"poll_time":<Integer_value>,

"svc_state_timestamp":<Double_value>,

"type_svc":<String_value>,

"appflowlog":<String_value>,

"svc_state":<String_value>,

"display_name":<String_value>,

"partition_name":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"save_config":<Boolean_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"graceful":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservice **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservice":[{

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"site_name":<String_value>,

"svr_name":<String_value>,

"id":<String_value>,

"svc_port":<Integer_value>,

"hostname":<String_value>,

"svc_name":<String_value>,
```

```
"publicip":<String_value>,

"svc_type":<String_value>,

"poll_time":<Integer_value>,

"svc_state_timestamp":<Double_value>,

"type_svc":<String_value>,

"appflowlog":<String_value>,

"svc_state":<String_value>,

"display_name":<String_value>,

"partition_name":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"save_config":<Boolean_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"graceful":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservice?action=disable;onerror=<String_value>

HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbservice": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_port":<Integer_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservice":[{

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"site_name":<String_value>,

"svr_name":<String_value>,

"id":<String_value>,

"svc_port":<Integer_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svc_type":<String_value>,

"poll_time":<Integer_value>,

"svc_state_timestamp":<Double_value>,

"type_svc":<String_value>,

"appflowlog":<String_value>,

"svc_state":<String_value>,

"display_name":<String_value>,

"partition_name":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"save_config":<Boolean_value>,
```

```
"svr_id":<String_value>,

"lb_id":<String_value>,

"graceful":<Boolean_value>}]}
```

ns_gslbservicegroup

Configuration for Citrix ADC GSLB Service Group Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
appflowlog	<String>	Read-write	AppFlow is enabled at GSLB servicegroup of GSLBVServer.
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
svc_grp_mem_port	<Integer>	Read-write	Service Group Member Port. Minimum value = 1 Maximum value =
svc_grp_name	<String>	Read-write	Service Group Name. Minimum length = 1 Maximum length = 100

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
svc_grp_mem_ip_address	<String>	Read-write	Service Group Member IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
lb_id	<String>	Read-write	LB ID.
vsvr_name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100
svr_name	<String>	Read-write	Server Member Name. Minimum length = 1 Maximum length = 100
delay	<Double>	Read-write	Time, in seconds, allocated to the Citrix ADC appliance for a graceful shutdown of the service.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
svr_id	<String>	Read-write	Server ID.
graceful	<Boolean>	Read-write	Graceful shutdown of Service..
svrstate	<String>	Read-only	Service State.
display_name	<String>	Read-only	Display Name.
svc_grp_effective_state	<String>	Read-only	Service Group Effective State.
partition_name	<String>	Read-only	Partition Name.
comment	<String>	Read-only	Comment.
svc_grp_mem_state	<String>	Read-only	Service Group Member State.
svc_grp_mem_type	<String>	Read-only	Service Group Member Type.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservicegroup?action=enable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbservicegroup": {

<b>"id":<String_value></b>,
```

```
"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_grp_mem_port":<Integer_value>,

"svr_name":<String_value>,

"svc_grp_name":<String_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"svc_grp_mem_ip_address":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservicegroup":[{

"svrstate":<String_value>,

"appflowlog":<String_value>,

"hostname":<String_value>,
```

```
"svc_grp_state_timestamp":<Double_value>,

"svc_grp_mem_port":<Integer_value>,

"svc_grp_name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"svc_grp_effective_state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"svc_grp_mem_ip_address":<String_value>,

"svc_grp_mem_state":<String_value>,

"id":<String_value>,

"svc_grp_mem_type":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"svr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"graceful":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservicegroup?action=poll;onerror=<String_value>

HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbservicegroup": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_grp_mem_port":<Integer_value>,

"svr_name":<String_value>,

"svc_grp_name":<String_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"svc_grp_mem_ip_address":<String_value>,

"lb_id":<String_value>,
```

```
"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservicegroup":[{

"svrstate":<String_value>,

"appflowlog":<String_value>,

"hostname":<String_value>,

"svc_grp_state_timestamp":<Double_value>,

"svc_grp_mem_port":<Integer_value>,

"svc_grp_name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"svc_grp_effective_state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"svc_grp_mem_ip_address":<String_value>,

"svc_grp_mem_state":<String_value>,

"id":<String_value>,
```

```
"svc_grp_mem_type":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"svr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"graceful":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservicegroup **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservicegroup":[{
```

```
"svrstate":<String_value>,

"appflowlog":<String_value>,

"hostname":<String_value>,

"svc_grp_state_timestamp":<Double_value>,

"svc_grp_mem_port":<Integer_value>,

"svc_grp_name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"svc_grp_effective_state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"svc_grp_mem_ip_address":<String_value>,

"svc_grp_mem_state":<String_value>,

"id":<String_value>,

"svc_grp_mem_type":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"svr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,
```

```
"graceful":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservicegroup?action=disable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbservicegroup": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_grp_mem_port":<Integer_value>,

"svr_name":<String_value>,

"svc_grp_name":<String_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,
```

```
"svc_grp_mem_ip_address":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservicegroup":[{

"svrstate":<String_value>,

"appflowlog":<String_value>,

"hostname":<String_value>,

"svc_grp_state_timestamp":<Double_value>,

"svc_grp_mem_port":<Integer_value>,

"svc_grp_name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"svc_grp_effective_state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"svc_grp_mem_ip_address":<String_value>,
```

```
"svc_grp_mem_state":<String_value>,

"id":<String_value>,

"svc_grp_mem_type":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"svr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"graceful":<Boolean_value>}]}
```

ns_gslbservicegroupmember_binding

Configuration for Citrix ADC GSLB Service Group Member Bindings resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip	<String>	Read-write	Server IP.
state	<String>	Read-write	state.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
weight	<String>	Read-write	Weight.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key.
servicegroupname	<String>	Read-write	Service Group Name.
svrstate	<String>	Read-write	Server State.
port	<Integer>	Read-write	Server Port.
servername	<String>	Read-write	ServerName.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
delay	<Double>	Read-write	Time, in seconds, allocated to the Citrix ADC appliance for a graceful shutdown of the service.
svc_grp_id	<String>	Read-write	Service Group ID.
graceful	<Boolean>	Read-write	Graceful shutdown of Service..
hostname	<String>	Read-only	Host name of the Managed device .
display_name	<String>	Read-only	Display Name.
comment	<String>	Read-only	Comment.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- ENABLE
- POLL

- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservicegroupmember_binding?action=enable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbservicegroupmember_binding": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"weight":<String_value>,

"servername":<String_value>,

"servicegroupname":<String_value>,

"graceful":<Boolean_value>,

"ip":<String_value>,

"delay":<Double_value>,

"state":<String_value>,

"svrstate":<String_value>,

"svc_grp_id":<String_value>,

"port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservicegroupmember_binding":[{

"ip":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"customserverid":<String_value>,

"weight":<String_value>,

"id":<String_value>,

"servicegroupname":<String_value>,

"svrstate":<String_value>,

"poll_time":<Integer_value>,

"port":<Integer_value>,

"display_name":<String_value>,

"serverid":<String_value>,

"servername":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"hash_id":<String_value>,

"save_config":<Boolean_value>,

"delay":<Double_value>,
```

```
"svc_grp_id":<String_value>,

"graceful":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservicegroupmember_binding?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbservicegroupmember_binding": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"weight":<String_value>,

"servername":<String_value>,

"servicegroupname":<String_value>,

"graceful":<Boolean_value>,

"ip":<String_value>,

"delay":<Double_value>,

"state":<String_value>,
```

```
"svrstate":<String_value>,

"svc_grp_id":<String_value>,

"port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservicegroupmember_binding":[{

"ip":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"customserverid":<String_value>,

"weight":<String_value>,

"id":<String_value>,

"servicegroupname":<String_value>,

"svrstate":<String_value>,

"poll_time":<Integer_value>,

"port":<Integer_value>,

"display_name":<String_value>,

"serverid":<String_value>,
```

```
"servername":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"hash_id":<String_value>,

"save_config":<Boolean_value>,

"delay":<Double_value>,

"svc_grp_id":<String_value>,

"graceful":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservicegroupmember_binding **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservicegroupmember_binding":[{

"ip":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"customserverid":<String_value>,

"weight":<String_value>,

"id":<String_value>,

"servicegroupname":<String_value>,

"svrstate":<String_value>,

"poll_time":<Integer_value>,

"port":<Integer_value>,

"display_name":<String_value>,

"serverid":<String_value>,

"servername":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"hash_id":<String_value>,

"save_config":<Boolean_value>,

"delay":<Double_value>,

"svc_grp_id":<String_value>,
```

```
"graceful":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbservicegroupmember_binding?action=disable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbservicegroupmember_binding": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"weight":<String_value>,

"servername":<String_value>,

"servicegroupname":<String_value>,

"graceful":<Boolean_value>,

"ip":<String_value>,

"delay":<Double_value>,

"state":<String_value>,

"svrstate":<String_value>,
```

```
"svc_grp_id":<String_value>,

"port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbservicegroupmember_binding":[{

"ip":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"customserverid":<String_value>,

"weight":<String_value>,

"id":<String_value>,

"servicegroupname":<String_value>,

"svrstate":<String_value>,

"poll_time":<Integer_value>,

"port":<Integer_value>,

"display_name":<String_value>,

"serverid":<String_value>,

"servername":<String_value>,
```

```
"comment":<String_value>,

"partition_name":<String_value>,

"hash_id":<String_value>,

"save_config":<Boolean_value>,

"delay":<Double_value>,

"svc_grp_id":<String_value>,

"graceful":<Boolean_value>}]}
```

ns_gslbvserver

Configuration for Citrix ADC GSLbvserver Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
user_managed	<Boolean>	Read-write	User Managed Idenfication.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
export_list	<String>	Read-write	Export List present for the given LB VServer.
id	<String>	Read-write	Id is system generated key.
vsvr_ip_address	<String>	Read-write	Vserver IP Address. Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	throughput value.
icalog	<String>	Read-write	ICA log.
appflow_policy_rule	<String>	Read-write	AppFlow policy rule present.
es4nslog	<String>	Read-write	ESNS enable.
transport_mode	<String>	Read-write	Transport type - IPFix or LogStream.
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
instance_license	<String>	Read-write	Feature License for Citrix ADC Instance, needs to be set while provisioning (standard, enterprise, platinum). Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
agent_list	<String>	Read-write	Agent List, on which traffic will flow.
export_option	<String>	Read-write	Export Options present for the given LB VServer.
appflowlog_effective	<String>	Read-write	Appflowlog effective (validating AppFlow config) enabled or disabled.
appflowlog	<String>	Read-write	Appflowlog enabled or disabled.
csvip_name	<String>	Read-write	CSVIP Name.
svc_id	<String>	Read-write	Service ID.
is_throughput_req	<Boolean>	Read-write	Throghput required.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
svc_name	<String>	Read-write	Service Name.
svc_grp_name	<String>	Read-write	Service Group Name.
svc_grp_id	<String>	Read-write	Service Group ID.
is_bot_policy_bound	<Boolean>	Read-write	Is Bot Policy bound to vserver..
csvip_id	<String>	Read-write	CSVIP ID.
app_security	<String>	Read-write	App Security to know whether bot and

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			appfw policies are bound..
is_appfw_policy_bound	<Boolean>	Read-write	Is Appfw Policy bound to vserver..
targetlbvserver	<String>	Read-only	Default Target LBVserver.
hits	<String>	Read-only	Traffic Domain.
license_type	<Integer>	Read-only	License Type : 0 - not licensed, 1 - manual, 2 - vpx, 3 - cpx, 4 - auto, 5 - cpx-sidecar/free.
state	<String>	Read-only	State of VServer.
cachetype	<String>	Read-only	Cache Type.
device_type	<String>	Read-only	Device Type : vpx, cpx, sdwan, blx, cpx-sidecar.
display_name	<String>	Read-only	Display Name of Vserver.
comment	<String>	Read-only	Vserver comment.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbvserver?action=enable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,
```

```
"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,
```

```
"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,
```

```
"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,
```

```
"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbvserver?action=poll;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,
```

```
"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,
```

```
"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,
```

```
"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbvserver **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,
```

```
"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,
```

```
"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbvserver?action=disable;onerror=<String_value>

HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,
```

```
"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbvserver":[{

"user_managed":<Boolean_value>,
```

```
"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,
```

```
"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,
```

```
"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

ns_gslbvserver_domain

Configuration for Citrix ADC GSLbvserver Domain Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	GSLB Domain Name.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
display_name	<String>	Read-only	Display Name.
state	<String>	Read-only	GSLB Vserver Domain State.
partition_name	<String>	Read-only	Partition Name.
servicetype	<String>	Read-only	GSLB service type.

Operations

(click to see Properties)

- POLL
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbvserver_domain?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_gslbvserver_domain": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbvserver_domain":[{

"name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"id":<String_value>,

"servicetype":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_gslbvserver_domain **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_gslbvserver_domain":[{

"name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"id":<String_value>,

"servicetype":<String_value>}]}
```

HAProxy

-

Load-Balancing

- ns_csvserver_policy_binding
- ns_lbvserver
- ns_lbvserver_policy_binding
- ns_lbvserver_service_binding
- ns_server
- ns_service
- ns_servicegroup
- ns_servicegroupmember_binding
- ns_services
- ns_vpnvserver_policy_binding
- ns_vserver
- ns_waf_learning_group

- ns_waf_relaxation

ns_csvserver_policy_binding

Configuration for Citrix ADC CSVServer Policy Binding Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
priority	<Integer>	Read-write	Priority of the Policy configured.
policy_name	<String>	Read-write	Policy Name. Minimum length = 1 Maximum length = 100
policy_type	<String>	Read-write	Type of the Policy configured.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
type	<String>	Read-write	Type of the binding configured.
vsvr_name	<String>	Read-only	VServer Name.
display_name	<String>	Read-only	Display Name.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_csvserver_policy_binding **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_csvserver_policy_binding":[{

"priority":<Integer_value>,

"vsvr_name":<String_value>,

"policy_name":<String_value>,

"policy_type":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"action":<String_value>,

"id":<String_value>,

"type":<String_value>}]}
```

ns_lbvserver

Configuration for Citrix ADC Lbvserver Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
user_managed	<Boolean>	Read-write	User Managed Idenfication.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100
export_list	<String>	Read-write	Export List present for the given LB VServer.
id	<String>	Read-write	Id is system generated key.
vsvr_ip_address	<String>	Read-write	Vserver IP Address. Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	throughput value.
icalog	<String>	Read-write	ICA log.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
appflow_policy_rule	<String>	Read-write	AppFlow policy rule present.
es4nslog	<String>	Read-write	ESNS enable.
transport_mode	<String>	Read-write	Transport type - IPFix or LogStream.
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
instance_license	<String>	Read-write	Feature License for Citrix ADC Instance, needs to be set while provisioning (standard, enterprise, platinum). Minimum length = 1 Maximum length = 64
agent_list	<String>	Read-write	Agent List, on which traffic will flow.
export_option	<String>	Read-write	Export Options present for the given LB VServer.
appflowlog_effective	<String>	Read-write	Appflowlog effective (validating AppFlow config) enabled or disabled.
appflowlog	<String>	Read-write	Appflowlog enabled or disabled.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
csvip_name	<String>	Read-write	CSVIP Name.
svc_id	<String>	Read-write	Service ID.
is_throughput_req	<Boolean>	Read-write	Throghput required.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
svc_name	<String>	Read-write	Service Name.
svc_grp_name	<String>	Read-write	Service Group Name.
svc_grp_id	<String>	Read-write	Service Group ID.
is_bot_policy_bound	<Boolean>	Read-write	Is Bot Policy bound to vserver..
csvip_id	<String>	Read-write	CSVIP ID.
app_security	<String>	Read-write	App Security to know whether bot and appfw policies are bound..
is_appfw_policy_bound	<Boolean>	Read-write	Is Appfw Policy bound to vserver..
targetlbvserver	<String>	Read-only	Default Target LBVserver.
hits	<String>	Read-only	Traffic Domain.
license_type	<Integer>	Read-only	License Type : 0 - not licensed, 1 - manual, 2 - vpx, 3 - cpx, 4 - auto, 5 - cpx-sidecar/free.
state	<String>	Read-only	State of VServer.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cachetype	<String>	Read-only	Cache Type.
device_type	<String>	Read-only	Device Type : vpx, cpx, sdwan, blx, cpx-sidecar.
display_name	<String>	Read-only	Display Name of Vserver.
comment	<String>	Read-only	Vserver comment.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_lbvserver?action=enable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_lbvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,
```

```
"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_lbvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,
```

```
"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,
```

```
"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_lbvserver?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_lbvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,
```

```
"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_lbvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,
```

```
"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,
```

```
"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,
```

```
"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_lbvserver **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_lbvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,
```

```
"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,
```

```
"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,
```

```
"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_lbvserver?action=disable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_lbvserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,
```

```
"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,
```

```
"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_lbvserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"hits":<String_value>,

"license_type":<Integer_value>,

"persistencetype":<String_value>,

"id":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"type_vserver":<String_value>,
```

```
"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"vsvr_max_users":<String_value>,

"es4nslog":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_id":<String_value>,

"parent_app_name":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"vsvr_type":<String_value>,

"vsvr_cur_users":<String_value>,

"vsvr_state_timestamp":<Double_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"managed":<Boolean_value>,

"instance_license":<String_value>,

"vsvr_state":<String_value>,

"state":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,
```

```
"export_option":<String_value>,

"poll_time":<Integer_value>,

"lbmethod":<String_value>,

"appflowlog_effective":<String_value>,

"activeservices":<String_value>,

"appflowlog":<String_value>,

"display_name":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"csvip_name":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_name":<String_value>,

"svc_grp_name":<String_value>,

"svc_grp_id":<String_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"app_security":<String_value>,

"is_appfw_policy_bound":<Boolean_value>}]}
```

ns_lbvserver_policy_binding

Configuration for Citrix ADC LBVServer Policy Binding Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
priority	<Integer>	Read-write	Priority of the Policy configured.
policy_name	<String>	Read-write	Policy Name. Minimum length = 1 Maximum length = 100
policy_type	<String>	Read-write	Type of the Policy configured.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
type	<String>	Read-write	Type of the binding configured.
vsvr_name	<String>	Read-only	VServer Name.
display_name	<String>	Read-only	Display Name.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_lbvserver_policy_binding **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_lbvserver_policy_binding":[{

"priority":<Integer_value>,

"vsvr_name":<String_value>,

"policy_name":<String_value>,

"policy_type":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"id":<String_value>,

"type":<String_value>}]}
```

ns_lbvserver_service_binding

Configuration for Citrix ADC LBVServer Services Binding Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
svc_name	<String>	Read-write	Service Name. Minimum length = 1 Maximum length = 100
id	<String>	Read-write	Id is system generated key.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
vsvr_name	<String>	Read-only	VServer Name.
partition_name	<String>	Read-only	Partition Name.
display_name	<String>	Read-only	Display Name.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_lbvserver_service_binding **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_lbvserver_service_binding":[{

"vsvr_name":<String_value>,

"partition_name":<String_value>,

"svc_name":<String_value>,

"poll_time":<Integer_value>,

"id":<String_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>}]}
```

ns_server

Configuration for Citrix ADC Server Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
svr_ip_address	<String>	Read-write	Server IP Address. Minimum length = 1 Maximum length = 64
name	<String>	Read-write	Server Name. Minimum length = 1 Maximum length = 100

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
ns_ip_address	<String>	Read-write	Server IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
delay	<Double>	Read-write	Time, in seconds, allocated to the Citrix ADC appliance for a graceful shutdown of the server.
graceful	<Boolean>	Read-write	Graceful shutdown of server..
display_name	<String>	Read-only	Display Name.
state	<String>	Read-only	Server state.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET

- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_server?action=enable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_server": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"save_config":<Boolean_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svr_ip_address":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_server":[{

"svr_ip_address":<String_value>,

"name":<String_value>,

"hostname":<String_value>,
```

```
"poll_time":<Integer_value>,

"display_name":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"id":<String_value>,

"save_config":<Boolean_value>,

"delay":<Double_value>,

"graceful":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_server?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_server": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"name":<String_value>,
```

```
"save_config":<Boolean_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svr_ip_address":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_server":[{

"svr_ip_address":<String_value>,

"name":<String_value>,

"hostname":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"id":<String_value>,

"save_config":<Boolean_value>,

"delay":<Double_value>,
```

```
"graceful":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_server **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_server":[{

"svr_ip_address":<String_value>,

"name":<String_value>,

"hostname":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,
```

```
"partition_name":<String_value>,

"id":<String_value>,

"save_config":<Boolean_value>,

"delay":<Double_value>,

"graceful":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_server?action=disable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_server": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"name":<String_value>,

"save_config":<Boolean_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,
```

```
"svr_ip_address":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_server":[{

"svr_ip_address":<String_value>,

"name":<String_value>,

"hostname":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"id":<String_value>,

"save_config":<Boolean_value>,

"delay":<Double_value>,

"graceful":<Boolean_value>}]}
```

ns_service

Configuration for Citrix ADC Services Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
svc_ip_address	<String>	Read-write	Service IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
svc_port	<Integer>	Read-write	Service Port. Minimum value = 1 Maximum value =
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
svc_name	<String>	Read-write	Service Name. Minimum length = 1 Maximum length = 100
publicip	<String>	Read-write	Public IP Address. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
appflowlog	<String>	Read-write	AppFlow is enabled at service of LBVServer.
vsvr_name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100
delay	<Double>	Read-write	Time, in seconds, allocated to the Citrix ADC appliance for a graceful shutdown of the service.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
svr_id	<String>	Read-write	Server ID.
lb_id	<String>	Read-write	LB ID.
graceful	<Boolean>	Read-write	Graceful shutdown of Service..
site_name	<String>	Read-only	Site Name.
svr_name	<String>	Read-only	Server Name associated with service.
svc_type	<String>	Read-only	Service Protocol (HTTP|FTP).
poll_time	<Integer>	Read-only	Last Polling Time.
svc_state	<String>	Read-only	Service State.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
display_name	<String>	Read-only	Display Name.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_service?action=enable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_service": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_port":<Integer_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,
```

```
"svc_name":<String_value>,

"publicip":<String_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_service":[{

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"site_name":<String_value>,

"svr_name":<String_value>,

"id":<String_value>,

"svc_port":<Integer_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svc_type":<String_value>,

"poll_time":<Integer_value>,
```

```
"svc_state_timestamp":<Double_value>,

"type_svc":<String_value>,

"appflowlog":<String_value>,

"svc_state":<String_value>,

"display_name":<String_value>,

"partition_name":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"save_config":<Boolean_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"graceful":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_service?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_service": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_port":<Integer_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_service":[{

"ns_ip_address":<String_value>,
```

```
"svc_ip_address":<String_value>,

"site_name":<String_value>,

"svr_name":<String_value>,

"id":<String_value>,

"svc_port":<Integer_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svc_type":<String_value>,

"poll_time":<Integer_value>,

"svc_state_timestamp":<Double_value>,

"type_svc":<String_value>,

"appflowlog":<String_value>,

"svc_state":<String_value>,

"display_name":<String_value>,

"partition_name":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"save_config":<Boolean_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"graceful":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_service **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_service":[{

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"site_name":<String_value>,

"svr_name":<String_value>,

"id":<String_value>,

"svc_port":<Integer_value>,

"hostname":<String_value>,

"svc_name":<String_value>,
```

```
"publicip":<String_value>,

"svc_type":<String_value>,

"poll_time":<Integer_value>,

"svc_state_timestamp":<Double_value>,

"type_svc":<String_value>,

"appflowlog":<String_value>,

"svc_state":<String_value>,

"display_name":<String_value>,

"partition_name":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"save_config":<Boolean_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"graceful":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_service?action=disable;onerror=<String_value>

HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_service": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_port":<Integer_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_service":[{

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"site_name":<String_value>,

"svr_name":<String_value>,

"id":<String_value>,

"svc_port":<Integer_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svc_type":<String_value>,

"poll_time":<Integer_value>,

"svc_state_timestamp":<Double_value>,

"type_svc":<String_value>,

"appflowlog":<String_value>,

"svc_state":<String_value>,

"display_name":<String_value>,

"partition_name":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"save_config":<Boolean_value>,
```

```
"svr_id":<String_value>,

"lb_id":<String_value>,

"graceful":<Boolean_value>}]}
```

ns_servicegroup

Configuration for Citrix ADC Service Group Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
appflowlog	<String>	Read-write	AppFlow is enabled at servicegroup of LBVServer.
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
svc_grp_mem_port	<Integer>	Read-write	Service Group Member Port. Minimum value = 1 Maximum value =
svc_grp_name	<String>	Read-write	Service Group Name. Minimum length = 1 Maximum length = 100
ns_ip_address	<String>	Read-write	Citrix ADC IP Address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
svc_grp_mem_ip_address	<String>	Read-write	Service Group Member IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
lb_id	<String>	Read-write	LB ID.
vsvr_name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100
svr_name	<String>	Read-write	Server Member Name. Minimum length = 1 Maximum length = 100
delay	<Double>	Read-write	Time, in seconds, allocated to the Citrix ADC appliance for a graceful shutdown of the service.
svr_id	<String>	Read-write	Server ID.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
graceful	<Boolean>	Read-write	Graceful shutdown of Service..
svrstate	<String>	Read-only	Service State.
display_name	<String>	Read-only	Display Name.
svc_grp_effective_state	<String>	Read-only	Service Group Effective State.
partition_name	<String>	Read-only	Partition Name.
comment	<String>	Read-only	Comment.
svc_grp_mem_state	<String>	Read-only	Service Group Member State.
svc_grp_mem_type	<String>	Read-only	Service Group Member Protocol Type.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_servicegroup?action=enable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_servicegroup": {

<b>"id":<String_value></b>,
```

```
"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_grp_mem_port":<Integer_value>,

"svr_name":<String_value>,

"svc_grp_name":<String_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"svc_grp_mem_ip_address":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_servicegroup":[{

"svrstate":<String_value>,

"appflowlog":<String_value>,

"hostname":<String_value>,
```

```
"svc_grp_state_timestamp":<Double_value>,

"svc_grp_mem_port":<Integer_value>,

"svc_grp_name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"svc_grp_effective_state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"svc_grp_mem_ip_address":<String_value>,

"svc_grp_mem_state":<String_value>,

"id":<String_value>,

"svc_grp_mem_type":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"svr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"graceful":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_servicegroup?action=poll;onerror=<String_value>

HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_servicegroup": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_grp_mem_port":<Integer_value>,

"svr_name":<String_value>,

"svc_grp_name":<String_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"svc_grp_mem_ip_address":<String_value>,

"lb_id":<String_value>,
```

```
"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_servicegroup":[{

"svrstate":<String_value>,

"appflowlog":<String_value>,

"hostname":<String_value>,

"svc_grp_state_timestamp":<Double_value>,

"svc_grp_mem_port":<Integer_value>,

"svc_grp_name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"svc_grp_effective_state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"svc_grp_mem_ip_address":<String_value>,

"svc_grp_mem_state":<String_value>,

"id":<String_value>,
```

```
"svc_grp_mem_type":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"svr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"graceful":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_servicegroup **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_servicegroup":[{
```

```
"svrstate":<String_value>,

"appflowlog":<String_value>,

"hostname":<String_value>,

"svc_grp_state_timestamp":<Double_value>,

"svc_grp_mem_port":<Integer_value>,

"svc_grp_name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"svc_grp_effective_state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"svc_grp_mem_ip_address":<String_value>,

"svc_grp_mem_state":<String_value>,

"id":<String_value>,

"svc_grp_mem_type":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"svr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,
```

```
"graceful":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_servicegroup?action=disable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_servicegroup": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_grp_mem_port":<Integer_value>,

"svr_name":<String_value>,

"svc_grp_name":<String_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,
```

```
"svc_grp_mem_ip_address":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_servicegroup":[{

"svrstate":<String_value>,

"appflowlog":<String_value>,

"hostname":<String_value>,

"svc_grp_state_timestamp":<Double_value>,

"svc_grp_mem_port":<Integer_value>,

"svc_grp_name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"svc_grp_effective_state":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"svc_grp_mem_ip_address":<String_value>,
```

```
"svc_grp_mem_state":<String_value>,

"id":<String_value>,

"svc_grp_mem_type":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"svr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"graceful":<Boolean_value>}]}
```

ns_servicegroupmember_binding

Configuration for Citrix ADC Service Group Member Bindings resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip	<String>	Read-write	Server IP address associated with this servicegroup.
state	<String>	Read-write	state of the servicegroup.
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
id	<String>	Read-write	Id is system generated key.
servicegroupname	<String>	Read-write	Service Group Name.
svrstate	<String>	Read-write	Server State.
port	<Integer>	Read-write	Server Port.
servername	<String>	Read-write	ServerName associated with this binding.
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
delay	<Double>	Read-write	Time, in seconds, allocated to the Citrix ADC appliance for a graceful shutdown of the service.
svc_grp_id	<String>	Read-write	Service Group ID.
graceful	<Boolean>	Read-write	Graceful shutdown of Service..
hostname	<String>	Read-only	Host name of the Managed device .
display_name	<String>	Read-only	Display Name for the binding.
comment	<String>	Read-only	Comment.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_servicegroupmember_binding?action=enable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_servicegroupmember_binding": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"servername":<String_value>,

"servicegroupname":<String_value>,

"graceful":<Boolean_value>,

"ip":<String_value>,

"delay":<Double_value>,

"state":<String_value>,

"svrstate":<String_value>,

"svc_grp_id":<String_value>,

"port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_servicegroupmember_binding":[{

"ip":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"customserverid":<String_value>,

"weight":<String_value>,

"id":<String_value>,

"servicegroupname":<String_value>,

"svrstate":<String_value>,

"poll_time":<Integer_value>,

"port":<Integer_value>,

"display_name":<String_value>,

"serverid":<String_value>,

"servername":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"hash_id":<String_value>,
```

```
"save_config":<Boolean_value>,

"delay":<Double_value>,

"svc_grp_id":<String_value>,

"graceful":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_servicegroupmember_binding?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_servicegroupmember_binding": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"servername":<String_value>,

"servicegroupname":<String_value>,

"graceful":<Boolean_value>,

"ip":<String_value>,

"delay":<Double_value>,
```

```
"state":<String_value>,

"svrstate":<String_value>,

"svc_grp_id":<String_value>,

"port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_servicegroupmember_binding":[{

"ip":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"customserverid":<String_value>,

"weight":<String_value>,

"id":<String_value>,

"servicegroupname":<String_value>,

"svrstate":<String_value>,

"poll_time":<Integer_value>,

"port":<Integer_value>,

"display_name":<String_value>,
```

```
"serverid":<String_value>,

"servername":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"hash_id":<String_value>,

"save_config":<Boolean_value>,

"delay":<Double_value>,

"svc_grp_id":<String_value>,

"graceful":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_servicegroupmember_binding **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_servicegroupmember_binding":[{

"ip":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"customserverid":<String_value>,

"weight":<String_value>,

"id":<String_value>,

"servicegroupname":<String_value>,

"svrstate":<String_value>,

"poll_time":<Integer_value>,

"port":<Integer_value>,

"display_name":<String_value>,

"serverid":<String_value>,

"servername":<String_value>,

"comment":<String_value>,

"partition_name":<String_value>,

"hash_id":<String_value>,

"save_config":<Boolean_value>,

"delay":<Double_value>,

"svc_grp_id":<String_value>,
```

```
"graceful":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_servicegroupmember_binding?action=disable;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_servicegroupmember_binding": {

<b>"id":<String_value></b>,

"ns_ip_address":<String_value>,

"save_config":<Boolean_value>,

"servername":<String_value>,

"servicegroupname":<String_value>,

"graceful":<Boolean_value>,

"ip":<String_value>,

"delay":<Double_value>,

"state":<String_value>,

"svrstate":<String_value>,

"svc_grp_id":<String_value>,
```

```
"port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_servicegroupmember_binding":[{

"ip":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"customserverid":<String_value>,

"weight":<String_value>,

"id":<String_value>,

"servicegroupname":<String_value>,

"svrstate":<String_value>,

"poll_time":<Integer_value>,

"port":<Integer_value>,

"display_name":<String_value>,

"serverid":<String_value>,

"servername":<String_value>,

"comment":<String_value>,
```

```
"partition_name":<String_value>,

"hash_id":<String_value>,

"save_config":<Boolean_value>,

"delay":<Double_value>,

"svc_grp_id":<String_value>,

"graceful":<Boolean_value>}]}
```

ns_services

Configuration for Citrix ADC Services Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
svc_port	<Integer>	Read-write	Service Port. Minimum value = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum value =
appflowlog	<String>	Read-write	AppFlow is enabled at service of LBVServer.
svc_ip_address	<String>	Read-write	Service IP Address. Minimum length = 1 Maximum length = 64
svc_name	<String>	Read-write	Service Name. Minimum length = 1 Maximum length = 100
publicip	<String>	Read-write	Public IP Address. Minimum length = 1 Maximum length = 64
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
lb_id	<String>	Read-write	LB ID.
vsvr_name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100
delay	<Double>	Read-write	Time, in seconds, allocated to the Citrix ADC appliance for a graceful shutdown of the service.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
svr_id	<String>	Read-write	Server ID.
graceful	<Boolean>	Read-write	Graceful shutdown of Service..
svr_name	<String>	Read-only	Server Name associated with service.
site_name	<String>	Read-only	Site Name.
svc_state	<String>	Read-only	Service State.
poll_time	<Integer>	Read-only	Last Polling Time.
svc_type	<String>	Read-only	Service Protocol (HTTP|FTP).
display_name	<String>	Read-only	Display Name.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_services?action=enable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_services": {

<b>"id":<String_value></b>,
```

```
"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_port":<Integer_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_services":[{

"svr_name":<String_value>,

"hostname":<String_value>,

"ns_ip_address":<String_value>,
```

```
"id":<String_value>,

"site_name":<String_value>,

"svc_port":<Integer_value>,

"svc_state":<String_value>,

"appflowlog":<String_value>,

"poll_time":<Integer_value>,

"svc_type":<String_value>,

"display_name":<String_value>,

"svc_ip_address":<String_value>,

"partition_name":<String_value>,

"svc_name":<String_value>,

"svc_state_timestamp":<Double_value>,

"type_svc":<String_value>,

"publicip":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"graceful":<Boolean_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_services?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_services": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_port":<Integer_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svc_name":<String_value>,

"publicip":<String_value>,

"svr_id":<String_value>,

"lb_id":<String_value>,
```

```
"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_services":[{

"svr_name":<String_value>,

"hostname":<String_value>,

"ns_ip_address":<String_value>,

"id":<String_value>,

"site_name":<String_value>,

"svc_port":<Integer_value>,

"svc_state":<String_value>,

"appflowlog":<String_value>,

"poll_time":<Integer_value>,

"svc_type":<String_value>,

"display_name":<String_value>,

"svc_ip_address":<String_value>,

"partition_name":<String_value>,

"svc_name":<String_value>,

"svc_state_timestamp":<Double_value>,
```

```
"type_svc":<String_value>,

"publicip":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"graceful":<Boolean_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_services **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_services":[{
```

```
"svr_name":<String_value>,

"hostname":<String_value>,

"ns_ip_address":<String_value>,

"id":<String_value>,

"site_name":<String_value>,

"svc_port":<Integer_value>,

"svc_state":<String_value>,

"appflowlog":<String_value>,

"poll_time":<Integer_value>,

"svc_type":<String_value>,

"display_name":<String_value>,

"svc_ip_address":<String_value>,

"partition_name":<String_value>,

"svc_name":<String_value>,

"svc_state_timestamp":<Double_value>,

"type_svc":<String_value>,

"publicip":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,
```

```
"graceful":<Boolean_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_services?action=disable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_services": {

<b>"id":<String_value></b>,

"vsvr_name":<String_value>,

"ns_ip_address":<String_value>,

"svc_ip_address":<String_value>,

"save_config":<Boolean_value>,

"svc_port":<Integer_value>,

"graceful":<Boolean_value>,

"hostname":<String_value>,

"delay":<Double_value>,

"svc_name":<String_value>,

"publicip":<String_value>,
```

```
"svr_id":<String_value>,

"lb_id":<String_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_services":[{

"svr_name":<String_value>,

"hostname":<String_value>,

"ns_ip_address":<String_value>,

"id":<String_value>,

"site_name":<String_value>,

"svc_port":<Integer_value>,

"svc_state":<String_value>,

"appflowlog":<String_value>,

"poll_time":<Integer_value>,

"svc_type":<String_value>,

"display_name":<String_value>,

"svc_ip_address":<String_value>,

"partition_name":<String_value>,
```

```
"svc_name":<String_value>,

"svc_state_timestamp":<Double_value>,

"type_svc":<String_value>,

"publicip":<String_value>,

"save_config":<Boolean_value>,

"lb_id":<String_value>,

"vsvr_name":<String_value>,

"delay":<Double_value>,

"svr_id":<String_value>,

"graceful":<Boolean_value>}]}
```

ns_vpnvserver_policy_binding

Configuration for Citrix ADC VPNVServer Policy Binding Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
priority	<Integer>	Read-write	Priority of the Policy configured.
policy_name	<String>	Read-write	Policy Name. Minimum length = 1 Maximum length = 100
ns_ip_address	<String>	Read-write	Citrix ADC IP Address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
type	<String>	Read-write	Type of the Policy configured.
vsvr_name	<String>	Read-only	VServer Name.
display_name	<String>	Read-only	Display Name.
partition_name	<String>	Read-only	Partition Name.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_vpnvserver_policy_binding **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vpnvserver_policy_binding":[{

"priority":<Integer_value>,

"vsvr_name":<String_value>,
```

```
"policy_name":<String_value>,

"poll_time":<Integer_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>,

"partition_name":<String_value>,

"id":<String_value>,

"type":<String_value>}]}
```

ns_vserver

Configuration for Citrix ADC Lbvserver Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
user_managed	<Boolean>	Read-write	User Managed Idenfication.
hostname	<String>	Read-write	Hostname of the managed device. Minimum length = 1 Maximum length = 256
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
agent_list	<String>	Read-write	Agent List, on which traffic will flow.
id	<String>	Read-write	Id is system generated key.
export_option	<String>	Read-write	Export Options present for the given LB VServer.
name	<String>	Read-write	Vserver Name. Minimum length = 1 Maximum length = 100
export_list	<String>	Read-write	Export List present for the given LB VServer.
vsvr_ip_address	<String>	Read-write	Vserver IP Address. Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	throughput value.
appflowlog_effective	<String>	Read-write	Appflowlog effective (validating AppFlow config) enabled or disabled.
icalog	<String>	Read-write	ICA log.
appflowlog	<String>	Read-write	Appflowlog enabled or disabled.
appflow_policy_rule	<String>	Read-write	AppFlow policy rule present.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
es4nslog	<String>	Read-write	ESNS enable.
transport_mode	<String>	Read-write	Transport type - IPFix or LogStream.
instance_license	<String>	Read-write	Feature License for Citrix ADC Instance, needs to be set while provisioning (standard, enterprise, platinum). Minimum length = 1 Maximum length = 64
save_config	<Boolean>	Read-write	Save configuration after enable/disable.
is_throughput_req	<Boolean>	Read-write	Throghput required.
is_bot_policy_bound	<Boolean>	Read-write	Is Bot Policy bound to vserver..
svc_grp_name	<String>	Read-write	Service Group Name.
csvip_id	<String>	Read-write	CSVIP ID.
svc_grp_id	<String>	Read-write	Service Group ID.
app_security	<String>	Read-write	App Security to know whether bot and appfw policies are bound..
csvip_name	<String>	Read-write	CSVIP Name.
is_appfw_policy_bound	<Boolean>	Read-write	Is Appfw Policy bound to vserver..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
svc_name	<String>	Read-write	Service Name.
svc_id	<String>	Read-write	Service ID.
state	<String>	Read-only	State of VServer.
cachetype	<String>	Read-only	Cache Type.
device_type	<String>	Read-only	Device Type : vpx, cpx, sdwan, blx, cpx-sidecar.
targetlbvserver	<String>	Read-only	Default Target LBVserver.
hits	<String>	Read-only	Traffic Domain.
license_type	<Integer>	Read-only	License Type : 0 - not licensed, 1 - manual, 2 - vpx, 3 - cpx, 4 - auto, 5 - cpx-sidecar/free.
display_name	<String>	Read-only	Display Name of Vserver.
comment	<String>	Read-only	Vserver comment.

Operations

(click to see Properties)

- ENABLE
- POLL
- GET
- DISABLE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

ENABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_vserver?action=enable;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_vserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,
```

```
"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"vsvr_state":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,
```

```
"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"vsvr_type":<String_value>,

"id":<String_value>,

"vsvr_cur_users":<String_value>,

"export_option":<String_value>,

"name":<String_value>,

"poll_time":<Integer_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"vsvr_state_timestamp":<Double_value>,

"lbmethod":<String_value>,

"hits":<String_value>,

"appflowlog_effective":<String_value>,

"type_vserver":<String_value>,

"activeservices":<String_value>,

"icalog":<String_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,
```

```
"vsvr_health":<Integer_value>,

"vsvr_max_users":<String_value>,

"appflowlog":<String_value>,

"appflow_policy_rule":<String_value>,

"license_type":<Integer_value>,

"es4nslog":<String_value>,

"parent_app_id":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_name":<String_value>,

"display_name":<String_value>,

"managed":<Boolean_value>,

"persistencetype":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"transport_mode":<String_value>,

"instance_license":<String_value>,

"save_config":<Boolean_value>,

"is_throughput_req":<Boolean_value>,

"is_bot_policy_bound":<Boolean_value>,

"svc_grp_name":<String_value>,

"csvip_id":<String_value>,

"svc_grp_id":<String_value>,
```

```
"app_security":<String_value>,

"csvip_name":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"svc_name":<String_value>,

"svc_id":<String_value>}]}
```

POLL

URL: https://<MGMT-IP>/nitro/v2/config/ns_vserver?action=poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_vserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,
```

```
"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,

"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"vsvr_state":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"vsvr_type":<String_value>,

"id":<String_value>,

"vsvr_cur_users":<String_value>,

"export_option":<String_value>,

"name":<String_value>,

"poll_time":<Integer_value>,

"export_list":<String_value>,
```

```
"targetlbvserver":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"vsvr_state_timestamp":<Double_value>,

"lbmethod":<String_value>,

"hits":<String_value>,

"appflowlog_effective":<String_value>,

"type_vserver":<String_value>,

"activeservices":<String_value>,

"icalog":<String_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"vsvr_max_users":<String_value>,

"appflowlog":<String_value>,

"appflow_policy_rule":<String_value>,

"license_type":<Integer_value>,

"es4nslog":<String_value>,

"parent_app_id":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_name":<String_value>,

"display_name":<String_value>,
```

```
"managed":<Boolean_value>,

"persistencetype":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"transport_mode":<String_value>,

"instance_license":<String_value>,

"save_config":<Boolean_value>,

"is_throughput_req":<Boolean_value>,

"is_bot_policy_bound":<Boolean_value>,

"svc_grp_name":<String_value>,

"csvip_id":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"csvip_name":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"svc_name":<String_value>,

"svc_id":<String_value>}]}
```

GET

URL: https://<MGMT-IP>/nitro/v2/config/ns_vserver **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vserver":[{

"user_managed":<Boolean_value>,

"curclntconnections":<Double_value>,

"vsvr_state":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"vsvr_type":<String_value>,

"id":<String_value>,

"vsvr_cur_users":<String_value>,

"export_option":<String_value>,
```

```
"name":<String_value>,

"poll_time":<Integer_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"vsvr_state_timestamp":<Double_value>,

"lbmethod":<String_value>,

"hits":<String_value>,

"appflowlog_effective":<String_value>,

"type_vserver":<String_value>,

"activeservices":<String_value>,

"icalog":<String_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"vsvr_max_users":<String_value>,

"appflowlog":<String_value>,

"appflow_policy_rule":<String_value>,

"license_type":<Integer_value>,

"es4nslog":<String_value>,

"parent_app_id":<String_value>,
```

```
"cursrvrconnections":<Double_value>,

"parent_app_name":<String_value>,

"display_name":<String_value>,

"managed":<Boolean_value>,

"persistencetype":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"transport_mode":<String_value>,

"instance_license":<String_value>,

"save_config":<Boolean_value>,

"is_throughput_req":<Boolean_value>,

"is_bot_policy_bound":<Boolean_value>,

"svc_grp_name":<String_value>,

"csvip_id":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"csvip_name":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"svc_name":<String_value>,

"svc_id":<String_value>}]}
```

DISABLE

URL: https://<MGMT-IP>/nitro/v2/config/ns_vserver?action=disable;onerror=<String_value>

HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ns_vserver": {

<b>"id":<String_value></b>,

"user_managed":<Boolean_value>,

"ns_ip_address":<String_value>,

"csvip_name":<String_value>,

"name":<String_value>,

"export_list":<String_value>,

"svc_id":<String_value>,

"is_throughput_req":<Boolean_value>,

"save_config":<Boolean_value>,

"svc_grp_name":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"is_bot_policy_bound":<Boolean_value>,
```

```
"csvip_id":<String_value>,

"icalog":<String_value>,

"appflow_policy_rule":<String_value>,

"es4nslog":<String_value>,

"transport_mode":<String_value>,

"hostname":<String_value>,

"svc_name":<String_value>,

"instance_license":<String_value>,

"agent_list":<String_value>,

"export_option":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"appflowlog_effective":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"appflowlog":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_vserver":[{

"user_managed":<Boolean_value>,
```

```
"curclntconnections":<Double_value>,

"vsvr_state":<String_value>,

"hostname":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"cachetype":<String_value>,

"agent_list":<String_value>,

"device_type":<String_value>,

"vsvr_type":<String_value>,

"id":<String_value>,

"vsvr_cur_users":<String_value>,

"export_option":<String_value>,

"name":<String_value>,

"poll_time":<Integer_value>,

"export_list":<String_value>,

"targetlbvserver":<String_value>,

"vsvr_ip_address":<String_value>,

"throughput":<Double_value>,

"vsvr_state_timestamp":<Double_value>,

"lbmethod":<String_value>,

"hits":<String_value>,

"appflowlog_effective":<String_value>,
```

```
"type_vserver":<String_value>,

"activeservices":<String_value>,

"icalog":<String_value>,

"totalservices":<String_value>,

"vsvr_port":<Integer_value>,

"vsvr_health":<Integer_value>,

"vsvr_max_users":<String_value>,

"appflowlog":<String_value>,

"appflow_policy_rule":<String_value>,

"license_type":<Integer_value>,

"es4nslog":<String_value>,

"parent_app_id":<String_value>,

"cursrvrconnections":<Double_value>,

"parent_app_name":<String_value>,

"display_name":<String_value>,

"managed":<Boolean_value>,

"persistencetype":<String_value>,

"partition_name":<String_value>,

"comment":<String_value>,

"transport_mode":<String_value>,

"instance_license":<String_value>,

"save_config":<Boolean_value>,
```

```
"is_throughput_req":<Boolean_value>,

"is_bot_policy_bound":<Boolean_value>,

"svc_grp_name":<String_value>,

"csvip_id":<String_value>,

"svc_grp_id":<String_value>,

"app_security":<String_value>,

"csvip_name":<String_value>,

"is_appfw_policy_bound":<Boolean_value>,

"svc_name":<String_value>,

"svc_id":<String_value>}]}
```

ns_waf_learning_group

Configuration for Reference object to logically group a ADC and WAF Profile name for WAF Learning resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ctnsappname	<String>	Read-write	ctnsappname. Minimum length = 1 Maximum length = 32
name	<String>	Read-write	name. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 32
display_name	<String>	Read-write	display_name for Citrix ADC . Minimum length = 1 Maximum length = 128
ns_ip_address	<String>	Read-write	Citrix ADC IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_waf_learning_group **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_waf_learning_group **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_waf_learning_group resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_waf_learning_group **?pagesize=#no;pageno=#no** Use this query-parameter to get the ns_waf_learning_group resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_waf_learning_group **?count=yes** Use this query-parameter to get the count of ns_waf_learning_group resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_waf_learning_group":[{

"ctnsappname":<String_value>,

"name":<String_value>,

"display_name":<String_value>,

"ns_ip_address":<String_value>}]}
```

ns_waf_relaxation

Configuration for Reference object to pass for relaxing a WAF Learning engine learnt rule resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
field_type	<String>	Read-write	Field type to be relaxed. Maximum length = 512
status	<Boolean>	Read-write	Status is true when the rule is enabled and false when the rule is disabled.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
mode	<Integer>	Read-write	mode to be deployed as open/closed.
violation_location	<Integer>	Read-write	Violation Location.
violation_id	<Integer>	Read-write	Violation Category.
field_name	<String>	Read-write	Field name to be relaxed. Maximum length = 512
resourceid	<String>	Read-write	Resourceid generated for each relaxation rule.. Maximum length = 128
max_value	<Integer>	Read-write	Maximum length of the filed to be relaxed.
value_type	<String>	Read-write	Value type to be relaxed. Maximum length = 512
http_req_url	<String>	Read-write	HTTP Request URL to be relaxed. Maximum length = 2000
value_expr	<String>	Read-write	Value expression to be relaxed. Maximum length = 512
si_device_ip_address	<String>	Read-write	Citrix ADC IP Address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
min_value	<Integer>	Read-write	minimum length of the field to be relaxed.
comment	<String>	Read-write	Comment to be sent as payload in relaxation rule nitro. Maximum length = 255
profile_name	<String>	Read-write	profile_name. Minimum length = 1 Maximum length = 32
adm_learn_profile_name	<String>	Read-write	adm_learn_profile_name. Minimum length = 1 Maximum length = 32

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

GET (ALL)

URL: https://<MGMT-IP>/nitro/v2/config/ns_waf_relaxation **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ns_waf_relaxation **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ns_waf_relaxation resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ns_waf_relaxation **? pagesize=#no;pageno=#no** Use this query-parameter to get the ns_waf_relaxation resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ns_waf_relaxation **?count=yes** Use this query-parameter to get the count of ns_waf_relaxation resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ns_waf_relaxation":[{

"field_type":<String_value>,

"status":<Boolean_value>,

"mode":<Integer_value>,

"violation_location":<Integer_value>,

"violation_id":<Integer_value>,

"field_name":<String_value>,

"resourceid":<String_value>,

"max_value":<Integer_value>,

"value_type":<String_value>,

"http_req_url":<String_value>,

"value_expr":<String_value>,
```

```
"si_device_ip_address":<String_value>,

"min_value":<Integer_value>,

"comment":<String_value>,

"profile_name":<String_value>,

"adm_learn_profile_name":<String_value>}]}
```

PaloAlto-VM-Series

- panw
- panw_device_profile
- panw_init_config
- panw_vpx_image

panw

Configuration for Panw Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	Image Name, This parameter is used while provisioning VM Instance with XVA image, template_name is given priority if provided along with image_name. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.
vm_memory_total	<Double>	Read-write	Total Memory of VM Instance in MB. 2048MB, 5120MB.
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
throughput_limit	<Double>	Read-write	Throughput Limit in Mbps set for VM Instance.
vlan_id_0_1	<Integer>	Read-write	VLAN id for the management interface 0/1. Maximum value =
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128
description	<String>	Read-write	Description of managed device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 512
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
fips_partition_name	<String>	Read-write	FIPS Partition Name. Minimum length = 1 Maximum length = 128
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.
burst_priority	<Integer>	Read-write	Burst Priority of the VM Instance between 1 and 4.
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1 Maximum length = 64
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
is_managed	<Boolean>	Read-write	Is Managed.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device. Minimum length = 1 Maximum length = 128
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device. Minimum length = 1 Maximum length = 64
network_interfaces	<network_interface[]>	Read-write	Network Interfaces.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
last_updated_time	<Double>	Read-write	Last Updated Time.
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.
host_ip_address	<String>	Read-write	Host IPAddress where VM is provisioned. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key for all the managed devices.
ipv6_address	<String>	Read-write	IPv6 Address.
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
license_edition	<String>	Read-write	Edition of instance.
number_of_acu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
device_family	<String>	Read-write	Device Family. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..
la_mgmt	<Boolean>	Read-write	Bond consisting of management ports on VM Instance.
vlan_id_0_2	<Integer>	Read-write	VLAN id for the management interface 0/2. Maximum value =
number_of_ssl_cores	<Integer>	Read-write	Assign number of ssl virtual functions to VM Instance.
type	<String>	Read-write	Type of device, (Xen | NS). Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	Assign throughput in Mbps to VM Instance.
template_name	<String>	Read-write	Template Name, This parameter is used while provisioning VM Instance with template, template_name is given priority if provided along with image_name. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64
max_burst_throughput	<Double>	Read-write	Maximum burst throughput in Mbps of VM Instance.
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal network (not required to be set for SDWAN).
node_id	<String>	Read-write	Node identification of a device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1 Maximum length = 64
throughput_allocation_mode	<Integer>	Read-write	Throughput Allocation Mode: 0-Fixed, 1-Burst-able.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
number_of_scu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
number_of_cores	<Integer>	Read-write	Number of cores that are assigned to VM Instance.
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be A-B, and for Cluster it will be CLIP. Minimum length = 1 Maximum length = 128
std_bw_total	<Integer>	Read-write	Standard Bandwidth.
domain_name	<String>	Read-write	Domain name of VM Device. Minimum length = 1 Maximum length = 128
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1 Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.
vlan_1_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 1/2 on VM Instance. Maximum value =
vrid_list_ipv4_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV4 VMAC Generation.
vlan_1_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/7 on VM Instance. Maximum value =
vrid_list_ipv6_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV6 VMAC Generation.
vrid_list_ipv4_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV4 VMAC Generation.
vrid_list_ipv6_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV6 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_10_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/4 on VM Instance.
vlan_10_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/5 on VM Instance. Maximum value =
vrid_list_ipv6_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV6 VMAC Generation.
vrid_list_ipv6_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV6 VMAC Generation.
if_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/5 on VM Instance.
receiveuntagged_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			on 1/4 on VM Instance.
vrid_list_ipv6_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV6 VMAC Generation.
vrid_list_ipv6_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV6 VMAC Generation.
if_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/3 on VM Instance.
vrid_list_ipv6_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV6 VMAC Generation.
receiveuntagged_10 _4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/4 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV6 VMAC Generation.
vlan_10_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/6 on VM Instance. Maximum value =
if_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/5 on VM Instance.
is_new_crypto	<Boolean>	Read-write	True if number_of_acu/ number_of_scu are used, false if number_of_ssl_core s is used.
vrid_list_ipv6_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV6 VMAC Generation.
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
if_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/7 on VM Instance.
receiveuntagged_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/8 on VM Instance.
vrid_list_ipv6_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV6 VMAC Generation.
receiveuntagged_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/6 on VM Instance.
vlan_1_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/4 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/2 on VM Instance.
receiveuntagged_1_ 5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/5 on VM Instance.
receiveuntagged_10 _6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/6 on VM Instance.
receiveuntagged_10 _1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/1 on VM Instance.
vrid_list_ipv4_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV4 VMAC Generation.
vrid_list_ipv6_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 1/5 for IPV6 VMAC Generation.
vrid_list_ipv6_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV6 VMAC Generation.
vrid_list_ipv4_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV4 VMAC Generation.
receiveuntagged_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/2 on VM Instance.
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
vrid_list_ipv4_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/7 on VM Instance.
vlan_10_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/4 on VM Instance. Maximum value =
if_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/7 on VM Instance.
receiveuntagged_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/1 on VM Instance.
vrid_list_ipv6_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV6 VMAC Generation.
if_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtNetwork 10/1 on VM Instance.
if_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/8 on VM Instance.
l2_enabled	<Boolean>	Read-write	L2mode status of VM Instance.
vlan_10_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/7 on VM Instance. Maximum value =
vlan_10_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/1 on VM Instance. Maximum value =
receiveuntagged_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/2 on VM Instance.
reboot_vm_on_cpu_change	<Boolean>	Read-write	Reboot VMs on CPU change during resource allocation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_1_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/6 on VM Instance. Maximum value =
vrid_list_ipv4_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV4 VMAC Generation.
receiveuntagged_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/7 on VM Instance.
vrid_list_ipv4_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV4 VMAC Generation.
vlan_1_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/1 on VM Instance. Maximum value =
vrid_list_ipv6_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVRID List for Interface 10/7 for IPV6 VMAC Generation.
vlan_1_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/8 on VM Instance. Maximum value =
vlan_10_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/2 on VM Instance. Maximum value =
receiveuntagged_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/5 on VM Instance.
vrid_list_ipv4_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV4 VMAC Generation.
vrid_list_ipv4_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 10/6 for IPV4 VMAC Generation.
vrid_list_ipv4_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV4 VMAC Generation.
receiveuntagged_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/8 on VM Instance.
vrid_list_ipv4_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV4 VMAC Generation.
vlan_1_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/3 on VM Instance. Maximum value =
receiveuntagged_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			on 10/3 on VM Instance.
if_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/6 on VM Instance.
if_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/6 on VM Instance.
vlan_10_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/3 on VM Instance. Maximum value =
vlan_1_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/5 on VM Instance. Maximum value =
if_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/3 on VM Instance.
vrid_list_ipv4_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 10/5 for IPV4 VMAC Generation.
if_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/4 on VM Instance.
vrid_list_ipv4_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV4 VMAC Generation.
if_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/1 on VM Instance.
vrid_list_ipv4_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV4 VMAC Generation.
vlan_10_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/8 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/3 on VM Instance.
crypto_change_requires_reboot	<Boolean>	Read-write	True if the current changes made by user requires a reboot of the VM else false.
vrid_list_ipv4_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for IPV4 VMAC Generation.
if_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/8 on VM Instance.
vrid_list_ipv4_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV4 VMAC Generation.
vrid_list_ipv6_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			IPV6 VMAC Generation.
if_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/2 on VM Instance.
vrid_list_ipv6_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV6 VMAC Generation.
sdxtools_http_port	<Integer>	Read-only	Http Port number to communicate to SDXTools.
instance_state	<String>	Read-only	State of device, UP only if device accessible.
reason	<String>	Read-only	Reason of failure for this managed device.
virtual_functions	<String>	Read-only	Virtual Functions assigned to VM Instance depending on SDX platform.
disk_space	<Double>	Read-only	Show Disk Space (GB) available to VM Instance.
panw_cpu_usage	<Double>	Read-only	CPU Usage (%) of Panw Instance.
upsince	<String>	Read-only	Upsince of managed device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
domid	<Double>	Read-only	DOM Id assigned to VM Instance viewed in hypervisor.
cpu_core_pe	<String>	Read-only	Packet Engine cores assigned to VM Instance.
tenant_id	<String>	Read-only	Tenant ID.
panw_memory_usage	<Double>	Read-only	Memory Usage (%) of Panw Instance.
preferred_cpu_socket	<String>	Read-only	Preferred CPU Socket.
device_uuid	<String>	Read-only	Device UUID.
do_config	<Boolean>	Read-only	Do default config for managed device.
version	<String>	Read-only	Device Version.
host_id	<String>	Read-only	Host ID.
has_sdxtools	<Boolean>	Read-only	True if SDX Tools are installed on this VM.
vm_memory_used	<Double>	Read-only	Memory Used (MB) by VM Instance.
number_of_cpu	<Integer>	Read-only	Number of CPU that is assigned to VM Instance.
is_swg	<Boolean>	Read-only	Boolean to indicate whether a VM is SWG.
sysid	<String>	Read-only	System ID.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vm_state	<String>	Read-only	State of Virtual Machine (Running | Halted).
uuid	<String>	Read-only	UUID of VM Instance generated by Hypervisor.
cpu_core_mgmt	<String>	Read-only	Management CPU cores assigned to VM Instance.
ssl_virtual_functions	<String>	Read-only	SSL Virtual Functions assigned to VM Instance.
uptime	<String>	Read-only	Uptime of device.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.
assigned_cpu_socket	<String>	Read-only	Assigned CPU Socket 0 or 1.
sdxtools_public_key	<String>	Read-only	SDXTools HTTPS Public Key.
status	<String>	Read-only	Status of managed device.
vm_description	<String>	Read-only	Description of the VM instance.
systemname	<String>	Read-only	Device System Name.
mac_eth0	<String>	Read-only	MAC Address of eth0 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vm_rx	<Double>	Read-only	In Throughput of VM Instance in Mbps.
disk_allocation	<String>	Read-only	Disk allocation for VM Instance.
vm_tx	<Double>	Read-only	Out Throughput of VM Instance in Mbps.
vm_memory_free	<Double>	Read-only	Free Memory (MB) available in VM Instance.
vm_memory_usage	<Double>	Read-only	Memory Usage (%) of VM Instance.
sdxtools_version	<String>	Read-only	SDXTools version running on VM Instance.
vm_cpu_usage	<Double>	Read-only	CPU Usage (%) of VM Instance.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- REBOOT
- STOP
- ADD
- FORCE_REBOOT
- GET (ALL)
- GET
- DELETE
- UPDATE

- FORCE_STOP
- START

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

reboot

URL: https://<MGMT-IP>/nitro/v2/config/panw/id_value<String>?action=reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"panw": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

stop

URL: https://<MGMT-IP>/nitro/v2/config/panw/id_value<String>?action=stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"panw": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

add

URL: https://<MGMT-IP>/nitro/v2/config/panw?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"panw": {

<b>"ip_address":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,
```

```
"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,
```

```
"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,
```

```
"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,
```

```
"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,
```

```
"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,
```

```
"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,
```

```
"vrid_list_ipv4_10_3":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"max_burst_throughput":<Double_value>,

"if_1_3":<Boolean_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"std_bw_total":<Integer_value>,
```

```
"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"panw":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,
```

```
"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"panw_cpu_usage":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"panw_memory_usage":<Double_value>,

"preferred_cpu_socket":<String_value>,
```

```
"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,
```

```
"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,
```

```
"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,
```

```
"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,
```

```
"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,
```

```
"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,
```

```
"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,
```

```
"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,
```

```
"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,
```

```
"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

force_reboot

URL: https://<MGMT-IP>/nitro/v2/config/panw/id_value<String>?action=force_reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"panw": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/panw **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/panw **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of panw resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/panw **? pagesize=#no;pageno=#no** Use this query-parameter to get the panw resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/panw **?count=yes** Use this query-parameter to get the count of panw resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"panw":[{
```

```
"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"panw_cpu_usage":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,
```

```
"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"panw_memory_usage":<Double_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,
```

```
"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,
```

```
"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>
```

```
"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,
```

```
"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,
```

```
"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,
```

```
"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,
```

```
"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,
```

```
"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,
```

```
"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/panw/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"panw":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,
```

```
"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"panw_cpu_usage":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,
```

```
"panw_memory_usage":<Double_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,
```

```
"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,
```

```
"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,
```

```
"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,
```

```
"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,
```

```
"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,
```

```
"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,
```

```
"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,
```

```
"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,
```

```
"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/panw/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

update

URL: https://<MGMT-IP>/nitro/v2/config/panw/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"panw":{

<b>"id":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,
```

```
"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,
```

```
"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,
```

```
"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,
```

```
"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{
```

"port_name":<String_value>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

```
"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,
```

```
"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"max_burst_throughput":<Double_value>,

"if_1_3":<Boolean_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,
```

```
"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"panw":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,
```

```
"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"panw_cpu_usage":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"panw_memory_usage":<Double_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,
```

```
"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,
```

```
"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,
```

```
"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,
```

```
"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,
```

```
"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,
```

```
"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,
```

```
"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,
```

```
"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,
```

```
"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,
```

```
"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

force_stop

URL: https://<MGMT-IP>/nitro/v2/config/panw/id_value<String>?action=force_stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"panw": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

start

URL: https://<MGMT-IP>/nitro/v2/config/panw/id_value<String>?action=start;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"panw": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

panw_device_profile

Configuration for svm.panw_name_tooltip device profile resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Profile Name. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
svm_ns_comm	<String>	Read-write	Communication protocol (http or https) with Instances. Minimum length = 1 Maximum length = 10
use_global_setting_for_communication_with_ns	<Boolean>	Read-write	True, if the communication with Instance needs to be global and not device specific.
id	<String>	Read-write	Id is system generated key for all the device profiles.
type	<String>	Read-write	Profile Type, This must be with in specified supported instance types: blx,sdvanvw,ns,nssdx,cbwanopt,cpx. Minimum length = 1 Maximum length = 128
ns_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix ADC profiles.
password	<String>	Read-write	Instance credentials.Password for this profile.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 127
snmpsecuritylevel	<String>	Read-write	SNMP v3 security level for this profile.
username	<String>	Read-write	Instance credentials.Username provided in the profile will be used to contact the instance. Minimum length = 1 Maximum length = 127
snmpauthprotocol	<String>	Read-write	SNMP v3 auth protocol for this profile.
ssh_port	<String>	Read-write	SSH port to connect to the device.
snmpprivprotocol	<String>	Read-write	SNMP v3 priv protocol for this profile.
host_password	<String>	Read-write	Host Password for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpversion	<String>	Read-write	SNMP version for this profile.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
passphrase	<String>	Read-write	Passphrase with which private key is encrypted.
host_username	<String>	Read-write	Host User Name for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpsecurityname	<String>	Read-write	SNMP v3 security name for this profile. Maximum length = 31
ssl_private_key	<String>	Read-write	SSL Private Key for key based authentication.
ssl_cert	<String>	Read-write	SSL Certificate for certificate based authentication.
http_port	<Integer>	Read-write	HTTP port to connect to the device.
snmpcommunity	<String>	Read-write	SNMP community for this profile. Maximum length = 31
https_port	<Integer>	Read-write	HTTPS port to connect to the device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
max_wait_time_reboot	<String>	Read-write	Max waiting time to reboot Citrix ADC.
snmpprivpassword	<String>	Read-write	SNMP v3 priv password for this profile. Minimum length = 8 Maximum length = 31
cb_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix SD-WAN profiles.
snmpauthpassword	<String>	Read-write	SNMP v3 auth password for this profile. Minimum length = 8 Maximum length = 31
tenant_id	<String>	Read-only	Tenant Id of device profile.
is_default	<Boolean>	Read-only	True, if this profile is system generated and can not be deleted.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/panw_device_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"panw_device_profile": {

<b>"name":<String_value></b>,

<b>"password":<String_value></b>,

<b>"username":<String_value></b>,

"snmpsecuritylevel":<String_value>,

"svm_ns_comm":<String_value>,

"ssh_port":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"type":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,
```

```
"cb_profile_name":<String_value>,

"ns_profile_name":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"panw_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,
```

```
"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

```

```

delete

URL: https://<MGMT-IP>/nitro/v2/config/panw_device_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/panw_device_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/panw_device_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of panw_device_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/panw_device_profile **? pagesize=#no;pageno=#no** Use this query-parameter to get the panw_device_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/panw_device_profile **?count=yes** Use this query-parameter to get the count of panw_device_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"panw_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,
```

```
"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/panw_device_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"panw_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,
```

```
"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

panw_init_config

Configuration for Initial Configuration Parameters of panw resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
prop2	<String>	Read-write	Property 2 of panw initial Configuration.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
prop1	<String>	Read-write	Property 1 of panw initial Configuration.
ip_address	<String>	Read-write	IP Address of panw. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- ADD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/panw_init_config?onerror=<String_value> **HTTPS**
Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"panw_init_config": {

<b>"ip_address":<String_value></b>,

"prop2":<String_value>,

"prop1":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"panw_init_config":[{

"prop2":<String_value>,

"prop1":<String_value>,

"ip_address":<String_value>}]}
```

panw_vpx_image

Configuration for XVA images of Panw Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD

- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/panw_vpx_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/panw_vpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/panw_vpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of panw_vpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/panw_vpx_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the panw_vpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/panw_vpx_image **?count=yes** Use this query-parameter to get the count of panw_vpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"panw_vpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/panw_vpx_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"panw_vpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/panw_vpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

SECUREMATRIX-GSB

- smx_gsb
- smx_gsb_device_profile
- smx_gsb_downgrade
- smx_gsb_image
- smx_gsb_init_config
- smx_gsb_upgrade
- smx_gsb_vpx_image

smx_gsb

Configuration for SecureMatrix Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	Image Name, This parameter is used while provisioning VM Instance with XVA image, template_name is given priority if provided along with image_name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 128
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.
vm_memory_total	<Double>	Read-write	Total Memory of VM Instance in MB. 2048MB, 5120MB.
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
throughput_limit	<Double>	Read-write	Throughput Limit in Mbps set for VM Instance.
vlan_id_0_1	<Integer>	Read-write	VLAN id for the management interface 0/1. Maximum value =
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
description	<String>	Read-write	Description of managed device. Minimum length = 1 Maximum length = 512
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
fips_partition_name	<String>	Read-write	FIPS Partition Name. Minimum length = 1 Maximum length = 128
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.
burst_priority	<Integer>	Read-write	Burst Priority of the VM Instance between 1 and 4.
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
is_managed	<Boolean>	Read-write	Is Managed.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device. Minimum length = 1 Maximum length = 128
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
network_interfaces	<network_interface[]>	Read-write	Network Interfaces.
last_updated_time	<Double>	Read-write	Last Updated Time.
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.
host_ip_address	<String>	Read-write	Host IPAddress where VM is provisioned. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key for all the managed devices.
ipv6_address	<String>	Read-write	IPv6 Address.
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
license_edition	<String>	Read-write	Edition of instance.
number_of_acu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
device_family	<String>	Read-write	Device Family.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..
la_mgmt	<Boolean>	Read-write	Bond consisting of management ports on VM Instance.
vlan_id_0_2	<Integer>	Read-write	VLAN id for the management interface 0/2. Maximum value =
number_of_ssl_cores	<Integer>	Read-write	Assign number of ssl virtual functions to VM Instance.
type	<String>	Read-write	Type of device, (Xen | NS). Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	Assign throughput in Mbps to VM Instance.
template_name	<String>	Read-write	Template Name, This parameter is used while provisioning VM Instance with template, template_name is given priority if provided along with image_name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 128
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64
max_burst_throughput	<Double>	Read-write	Maximum burst throughput in Mbps of VM Instance.
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal network (not required to be set for SDWAN).
node_id	<String>	Read-write	Node identification of a device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1 Maximum length = 64
throughput_allocation_mode	<Integer>	Read-write	Throughput Allocation Mode: 0-Fixed, 1-Burst-able.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
number_of_scu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
number_of_cores	<Integer>	Read-write	Number of cores that are assigned to VM Instance.
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be A-B, and for Cluster it will be CLIP. Minimum length = 1 Maximum length = 128
std_bw_total	<Integer>	Read-write	Standard Bandwidth.
domain_name	<String>	Read-write	Domain name of VM Device. Minimum length = 1 Maximum length = 128
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1 Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.
vlan_1_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Network 1/2 on VM Instance. Maximum value =
vrid_list_ipv4_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV4 VMAC Generation.
vlan_1_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/7 on VM Instance. Maximum value =
vrid_list_ipv6_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV6 VMAC Generation.
vrid_list_ipv4_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV4 VMAC Generation.
vrid_list_ipv6_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/6 for IPV6 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_10_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/4 on VM Instance.
vlan_10_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/5 on VM Instance. Maximum value =
vrid_list_ipv6_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV6 VMAC Generation.
vrid_list_ipv6_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV6 VMAC Generation.
if_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/5 on VM Instance.
receiveuntagged_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			on 1/4 on VM Instance.
vrid_list_ipv6_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV6 VMAC Generation.
vrid_list_ipv6_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV6 VMAC Generation.
if_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/3 on VM Instance.
vrid_list_ipv6_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV6 VMAC Generation.
receiveuntagged_10 _4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/4 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV6 VMAC Generation.
vlan_10_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/6 on VM Instance. Maximum value =
if_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/5 on VM Instance.
is_new_crypto	<Boolean>	Read-write	True if number_of_acu/ number_of_scu are used, false if number_of_ssl_core s is used.
vrid_list_ipv6_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV6 VMAC Generation.
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
if_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/7 on VM Instance.
receiveuntagged_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/8 on VM Instance.
vrid_list_ipv6_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV6 VMAC Generation.
receiveuntagged_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/6 on VM Instance.
vlan_1_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/4 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/2 on VM Instance.
receiveuntagged_1_ 5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/5 on VM Instance.
receiveuntagged_10 _6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/6 on VM Instance.
receiveuntagged_10 _1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/1 on VM Instance.
vrid_list_ipv4_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV4 VMAC Generation.
vrid_list_ipv6_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 1/5 for IPV6 VMAC Generation.
vrid_list_ipv6_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV6 VMAC Generation.
vrid_list_ipv4_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV4 VMAC Generation.
receiveuntagged_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/2 on VM Instance.
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
vrid_list_ipv4_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/7 on VM Instance.
vlan_10_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/4 on VM Instance. Maximum value =
if_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/7 on VM Instance.
receiveuntagged_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/1 on VM Instance.
vrid_list_ipv6_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV6 VMAC Generation.
if_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtNetwork 10/1 on VM Instance.
if_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/8 on VM Instance.
l2_enabled	<Boolean>	Read-write	L2mode status of VM Instance.
vlan_10_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/7 on VM Instance. Maximum value =
vlan_10_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/1 on VM Instance. Maximum value =
receiveuntagged_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/2 on VM Instance.
reboot_vm_on_cpu_change	<Boolean>	Read-write	Reboot VMs on CPU change during resource allocation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_1_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/6 on VM Instance. Maximum value =
vrid_list_ipv4_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV4 VMAC Generation.
receiveuntagged_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/7 on VM Instance.
vrid_list_ipv4_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV4 VMAC Generation.
vlan_1_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/1 on VM Instance. Maximum value =
vrid_list_ipv6_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVRID List for Interface 10/7 for IPV6 VMAC Generation.
vlan_1_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/8 on VM Instance. Maximum value =
vlan_10_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/2 on VM Instance. Maximum value =
receiveuntagged_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/5 on VM Instance.
vrid_list_ipv4_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV4 VMAC Generation.
vrid_list_ipv4_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 10/6 for IPV4 VMAC Generation.
vrid_list_ipv4_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV4 VMAC Generation.
receiveuntagged_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/8 on VM Instance.
vrid_list_ipv4_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV4 VMAC Generation.
vlan_1_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/3 on VM Instance. Maximum value =
receiveuntagged_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			on 10/3 on VM Instance.
if_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/6 on VM Instance.
if_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/6 on VM Instance.
vlan_10_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/3 on VM Instance. Maximum value =
vlan_1_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/5 on VM Instance. Maximum value =
if_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/3 on VM Instance.
vrid_list_ipv4_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 10/5 for IPV4 VMAC Generation.
if_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/4 on VM Instance.
vrid_list_ipv4_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV4 VMAC Generation.
if_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/1 on VM Instance.
vrid_list_ipv4_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV4 VMAC Generation.
vlan_10_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/8 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/3 on VM Instance.
crypto_change_requires_reboot	<Boolean>	Read-write	True if the current changes made by user requires a reboot of the VM else false.
vrid_list_ipv4_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for IPV4 VMAC Generation.
if_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/8 on VM Instance.
vrid_list_ipv4_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV4 VMAC Generation.
vrid_list_ipv6_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			IPV6 VMAC Generation.
if_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/2 on VM Instance.
vrid_list_ipv6_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV6 VMAC Generation.
cron_state	<String>	Read-only	Cron Daemon State of SecureMatrix Instance.
sdxtools_http_port	<Integer>	Read-only	Http Port number to communicate to SDXTools.
instance_state	<String>	Read-only	State of device, UP only if device accessible.
webserver_state	<String>	Read-only	Webserver State of SecureMatrix Instance.
reason	<String>	Read-only	Reason of failure for this managed device.
virtual_functions	<String>	Read-only	Virtual Functions assigned to VM Instance depending on SDX platform.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
disk_space	<Double>	Read-only	Show Disk Space (GB) available to VM Instance.
upsince	<String>	Read-only	Upsince of managed device.
domid	<Double>	Read-only	DOM Id assigned to VM Instance viewed in hypervisor.
cpu_core_pe	<String>	Read-only	Packet Engine cores assigned to VM Instance.
ssh_state	<String>	Read-only	SSH Daemon State of SecureMatrix Instance.
tenant_id	<String>	Read-only	Tenant ID.
preferred_cpu_socket	<String>	Read-only	Preferred CPU Socket.
device_uuid	<String>	Read-only	Device UUID.
do_config	<Boolean>	Read-only	Do default config for managed device.
version	<String>	Read-only	Device Version.
host_id	<String>	Read-only	Host ID.
has_sdxtools	<Boolean>	Read-only	True if SDX Tools are installed on this VM.
vm_memory_used	<Double>	Read-only	Memory Used (MB) by VM Instance.
number_of_cpu	<Integer>	Read-only	Number of CPU that is assigned to VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_swg	<Boolean>	Read-only	Boolean to indicate whether a VM is SWG.
sysid	<String>	Read-only	System ID.
vm_state	<String>	Read-only	State of Virtual Machine (Running | Halted).
uuid	<String>	Read-only	UUID of VM Instance generated by Hypervisor.
cpu_core_mgmt	<String>	Read-only	Management CPU cores assigned to VM Instance.
ssl_virtual_functions	<String>	Read-only	SSL Virtual Functions assigned to VM Instance.
uptime	<String>	Read-only	Uptime of device.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.
assigned_cpu_socket	<String>	Read-only	Assigned CPU Socket 0 or 1.
sdxtools_public_key	<String>	Read-only	SDXTools HTTPS Public Key.
status	<String>	Read-only	Status of managed device.
smx_gsb_memory_usage	<Double>	Read-only	Memory Usage (%) of SecureMatrix Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vm_description	<String>	Read-only	Description of the VM instance.
systemname	<String>	Read-only	Device System Name.
mac_eth0	<String>	Read-only	MAC Address of eth0 on VM Instance.
vm_rx	<Double>	Read-only	In Throughput of VM Instance in Mbps.
disk_allocation	<String>	Read-only	Disk allocation for VM Instance.
vm_tx	<Double>	Read-only	Out Throughput of VM Instance in Mbps.
smx_gsb_cpu_usage	<Double>	Read-only	CPU Usage (%) of SecureMatrix Instance.
vm_memory_free	<Double>	Read-only	Free Memory (MB) available in VM Instance.
vm_memory_usage	<Double>	Read-only	Memory Usage (%) of VM Instance.
sdxtools_version	<String>	Read-only	SDXTools version running on VM Instance.
vm_cpu_usage	<Double>	Read-only	CPU Usage (%) of VM Instance.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- REBOOT
- STOP
- ADD
- FORCE_REBOOT
- GET (ALL)
- GET
- DELETE
- UPDATE
- FORCE_STOP
- START

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

reboot

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb/id_value<String>?action=reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smx_gsb": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

stop

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb/id_value<String>?action=stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smx_gsb": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

add

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smx_gsb": {

<b>"ip_address":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,
```

```
"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,
```

```
"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,
```

```
"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,
```

```
"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,
```

```
"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,
```

```
"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,
```

```
"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"max_burst_throughput":<Double_value>,

"if_1_3":<Boolean_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,
```

```
"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"cron_state":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,
```

```
"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"webserver_state":<String_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,
```

```
"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"ssh_state":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,
```

```
"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,
```

```
"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,
```

```
"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,
```

```
"status":<String_value>,

"gateway":<String_value>,

"smx_gsb_memory_usage":<Double_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"smx_gsb_cpu_usage":<Double_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,
```

```
"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,
```

```
"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,
```

```
"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,
```

```
"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,
```

```
"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

force_reboot

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb/id_value<String>?action=force_reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smx_gsb": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/smx_gsb **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of smx_gsb resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/smx_gsb **? pagesize=#no;pageno=#no** Use this query-parameter to get the smx_gsb resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/smx_gsb **?count=yes** Use this query-parameter to get the count of smx_gsb resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"cron_state":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,
```

```
"webserver_state":<String_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"ssh_state":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,
```

```
"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{
```

```
"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,
```

```
"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,
```

```
"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"smx_gsb_memory_usage":<Double_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,
```

```
"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"smx_gsb_cpu_usage":<Double_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,
```

```
"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,
```

```
"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,
```

```
"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,
```

```
"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,
```

```
"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"cron_state":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"webserver_state":<String_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,
```

```
"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"ssh_state":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,
```

```
"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,
```

```
"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>
```

```
"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,
```

```
"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"smx_gsb_memory_usage":<Double_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,
```

```
"ip_address":<String_value>,

"smx_gsb_cpu_usage":<Double_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,
```

```
"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,
```

```
"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,
```

```
"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,
```

```
"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,
```

```
"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

update

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smx_gsb":{

<b>"id":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,
```

```
"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,
```

```
"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,
```

```
"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,
```

```
"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,
```

```
"netmask":<String_value>,

"ip_address":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,
```

```
"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"max_burst_throughput":<Double_value>,

"if_1_3":<Boolean_value>,

"config_type":<Integer_value>,
```

```
"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb":[{

"image_name":<String_value>,
```

```
"is_grace":<Boolean_value>,

"hostname":<String_value>,

"cron_state":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"webserver_state":<String_value>,

"name":<String_value>,

"reason":<String_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,
```

```
"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"ssh_state":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,
```

```
"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,
```

```
"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>
```

```
"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,
```

```
"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"smx_gsb_memory_usage":<Double_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"smx_gsb_cpu_usage":<Double_value>,

"throughput_allocation_mode":<Integer_value>,
```

```
"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,
```

```
"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,
```

```
"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,
```

```
"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,
```

```
"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

```
```

force_stop

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb/id_value<String>?action=force_stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smx_gsb": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

start

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb/id_value<String>?action=start;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smx_gsb": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

smx_gsb_device_profile

Configuration for svm.smx_gsb_name_tooltip device profile resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Profile Name. Minimum length = 1 Maximum length = 128
svm_ns_comm	<String>	Read-write	Communication protocol (http or https) with Instances. Minimum length = 1 Maximum length = 10

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
use_global_setting_for_communication_with_ns	<Boolean>	Read-write	True, if the communication with Instance needs to be global and not device specific.
id	<String>	Read-write	Id is system generated key for all the device profiles.
type	<String>	Read-write	Profile Type, This must be with in specified supported instance types: blx,sdvanvw,ns,nssdx,cbwanopt,cpx. Minimum length = 1 Maximum length = 128
ns_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix ADC profiles.
password	<String>	Read-write	Instance credentials.Password for this profile. Minimum length = 1 Maximum length = 127
snmpsecuritylevel	<String>	Read-write	SNMP v3 security level for this profile.
username	<String>	Read-write	Instance credentials.Username provided in the

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			profile will be used to contact the instance. Minimum length = 1 Maximum length = 127
snmpauthprotocol	<String>	Read-write	SNMP v3 auth protocol for this profile.
ssh_port	<String>	Read-write	SSH port to connect to the device.
snmpprivprotocol	<String>	Read-write	SNMP v3 priv protocol for this profile.
host_password	<String>	Read-write	Host Password for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpversion	<String>	Read-write	SNMP version for this profile.
passphrase	<String>	Read-write	Passphrase with which private key is encrypted.
host_username	<String>	Read-write	Host User Name for this profile.Used for BLX form factor of ADC. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 127
snmpsecurityname	<String>	Read-write	SNMP v3 security name for this profile. Maximum length = 31
ssl_private_key	<String>	Read-write	SSL Private Key for key based authentication.
ssl_cert	<String>	Read-write	SSL Certificate for certificate based authentication.
http_port	<Integer>	Read-write	HTTP port to connect to the device.
snmpcommunity	<String>	Read-write	SNMP community for this profile. Maximum length = 31
https_port	<Integer>	Read-write	HTTPS port to connect to the device.
max_wait_time_reboot	<String>	Read-write	Max waiting time to reboot Citrix ADC.
snmpprivpassword	<String>	Read-write	SNMP v3 priv password for this profile. Minimum length = 8 Maximum length = 31

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cb_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix SD-WAN profiles.
snmpauthpassword	<String>	Read-write	SNMP v3 auth password for this profile. Minimum length = 8 Maximum length = 31
tenant_id	<String>	Read-only	Tenant Id of device profile.
is_default	<Boolean>	Read-only	True, if this profile is system generated and can not be deleted.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_device_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smx_gsb_device_profile": {

<b>"name":<String_value></b>,
```

"password":<String_value>,

"username":<String_value>,

"snmpsecuritylevel":<String_value>,

"svm_ns_comm":<String_value>,

"ssh_port":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"type":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"ns_profile_name":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

```
"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,
```

```
"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_device_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_device_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/smx_gsb_device_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of smx_gsb_device_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/smx_gsb_device_profile **? pagesize=#no;pageno=#no** Use this query-parameter to get the smx_gsb_device_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/smx_gsb_device_profile **?count=yes** Use this query-parameter to get the count of smx_gsb_device_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,
```

```
"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_device_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,
```

```
"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

smx_gsb_downgrade

Configuration for Downgrade SMX GSB Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
smx_gsb_ip_address_arr	<String[]>	Read-write	List of SMX GSB IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- DOWNGRADE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

downgrade

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_downgrade?action=downgrade;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smx_gsb_downgrade": {

<b>"smx_gsb_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb_downgrade":[{

"smx_gsb_ip_address_arr":<String_value>}]}
```

smx_gsb_image

Configuration for SecureMatrix Build Files resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/smx_gsb_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of

smx_gsb_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/smx_gsb_image **?pagesize=#no;pageno=#no** Use this query-parameter to get the smx_gsb_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/smx_gsb_image **?count=yes** Use this query-parameter to get the count of smx_gsb_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,
```

```
"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

```
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/smx_gsb_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

smx_gsb_init_config

Configuration for Initial Configuration Parameters of smx_gsb resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
prop2	<String>	Read-write	Property 2 of smx_gsb initial Configuration.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
prop1	<String>	Read-write	Property 1 of smx_gsb initial Configuration.
ip_address	<String>	Read-write	IP Address of smx_gsb. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- ADD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_init_config?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smx_gsb_init_config": {

<b>"ip_address":<String_value></b>,

"prop2":<String_value>,

"prop1":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb_init_config":[{

"prop2":<String_value>,

"prop1":<String_value>,

"ip_address":<String_value>}]}
```

smx_gsb_upgrade

Configuration for Upgrade SMX GSB Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	SMX GSB Image Name. Minimum length = 1 Maximum length = 128
smx_gsb_ip_address_arr	<String[]>	Read-write	List of SMX GSB IP Address. Minimum length = 1 Maximum length = 64
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- UPGRADE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

upgrade

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_upgrade?action=upgrade;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smx_gsb_upgrade": {

<b>"image_name":<String_value></b>,

<b>"smx_gsb_ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb_upgrade":[{

"act_id":<String_value>,

"image_name":<String_value>,

"smx_gsb_ip_address_arr":<String_value>}]}
```

smx_gsb_vpx_image

Configuration for XVA images of SecureMatrix Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_vpx_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_vpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/smx_gsb_vpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of smx_gsb_vpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/smx_gsb_vpx_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the smx_gsb_vpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/smx_gsb_vpx_image **?count=yes** Use this query-parameter to get the count of smx_gsb_vpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb_vpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,
```

```
"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/smx_gsb_vpx_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smx_gsb_vpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,
```

```
"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/smx_gsb_vpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

System

- aa_adc_prediction_l2
- aaa_server
- activity_status
- af_waf_deploy_rules_l2
- allowed_clients
- audit_log
- autoselection_preference

- backup
- backup_component
- backup_external_storage
- backup_file
- backup_policy
- built_in_agent
- capacity
- cb_sb_image
- cb_sb_upgrade
- cb_sb_upgrade_monitor
- central_management_server
- cipher_config
- cipher_group
- cipher_suite
- clean_install_info
- cli_prompt
- counter_item
- csv_export
- current_hostname
- current_timezone
- device_backup
- device_backup_policy
- entity_map
- external_server
- jazz_license
- ldap_server
- login
- logout
- mail_attachment
- mail_profile

- managed_vserver
- metering_data
- metering_export
- metering_summary
- mps
- mps_disk_usage
- mps_feature
- mps_health
- mps_image
- mps_ssl_certkey
- mps_upgrade
- mpsgroup
- mpssession
- mpsuser
- ntp_param
- ntp_server
- ntp_sync
- password_policy
- platform_specs
- property_map
- prune_policy
- radius_server
- report_timezone
- restore
- sb_info
- sdx_autoreg
- sdx_backup_restore
- sdx_builtin_agent
- sdx_network_config
- sdxtools_image

- sdxtools_upgrade
- sent_mails
- sent_sms
- server_connection
- server_nodes
- service
- sms_profile
- sms_server
- smtp_server
- snmp_alarm_config
- snmp_info
- snmp_manager
- snmp_mib
- snmp_trap
- snmp_user
- snmp_view
- ssl_cert
- ssl_key
- ssl_settings
- static_route
- syslog_facility
- syslog_messages
- syslog_params
- syslog_persistant_stat_info
- syslog_purge_setting
- syslog_purge_setting_details
- syslog_server
- system_settings
- system_threshold
- tacacs_server

- task_command_log
- task_device_log
- task_log
- techsupport
- tenant
- tenant_auth_server
- tenant_company_info
- tenant_info
- tenant_network_pool
- tenant_provisioning_status
- tenant_system_info
- tenant_system_resource
- timezone_enum
- trap_forward_profile
- trust_preauthtoken
- user_lockout_policy

aa_adc_prediction_l2

Configuration for ADC Prediction data resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
window_end_time	<Double>	Read-write	Window End time..
window_start_time	<Double>	Read-write	Window Start time..
tenant_id	<String>	Read-write	tenant_id. Maximum length = 255
prediction_hour_id	<Double>	Read-write	Prediction Hour ID..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
model_run_time	<Double>	Read-write	model_run_time.
prediction_upbound	<Double>	Read-write	Prediction Upper bound..
ip_address	<String>	Read-write	ADC IP Address. Maximum length = 512
prediction_time	<Double>	Read-write	Prediction Time..
prediction_metric	<String>	Read-write	prediction_metric. Maximum length = 255
prediction_avg	<Double>	Read-write	Prediction Average.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/aa_adc_prediction_l2 **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/aa_adc_prediction_l2 **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of aa_adc_prediction_l2 resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/aa_adc_prediction_l2 **? pagesize=#no;pageno=#no** Use this query-parameter to get the aa_adc_prediction_l2 resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/aa_adc_prediction_l2 **?count=yes** Use this query-parameter to get the count of aa_adc_prediction_l2 resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aa_adc_prediction_l2":[{

"window_end_time":<Double_value>,

"window_start_time":<Double_value>,

"tenant_id":<String_value>,

"prediction_hour_id":<Double_value>,

"model_run_time":<Double_value>,

"prediction_upbound":<Double_value>,

"ip_address":<String_value>,

"prediction_time":<Double_value>,

"prediction_metric":<String_value>,

"prediction_avg":<Double_value>}]}
```

aaa_server

Configuration for AAA Server configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
external_servers	<external_server[]>	Read-write	List of external servers.
fallback_local_authentication	<Boolean>	Read-write	Enable local fallback authentication.
log_ext_group_info	<Boolean>	Read-write	Log external group info.
primary_server_name	<String>	Read-write	Name of primary server name. Minimum length = 1 Maximum length = 128
id	<String>	Read-write	Id is system generated key for all the servers.
primary_server_type	<String>	Read-write	Type of primary server. Supported types 1. LOCAL 2.RADIUS 3.LDAP 4.TACACS 5.KEYSTONE. Minimum length = 1 Maximum length = 32

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/aaa_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/aaa_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of aaa_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/aaa_server **?pagesize=#no;pageno=#no** Use this query-parameter to get the aaa_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/aaa_server **?count=yes** Use this query-parameter to get the count of aaa_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aaa_server":[{

"external_servers":[{

"priority":<Integer_value>,

"parent_name":<String_value>,

"external_server_type":<String_value>,

"external_server_name":<String_value>,

"id":<String_value>,

"parent_id":<String_value>}],
```

```
"secondary_server_name":<String_value>,

"fallback_local_authentication":<Boolean_value>,

"log_ext_group_info":<Boolean_value>,

"enable_secondary_server":<Boolean_value>,

"primary_server_name":<String_value>,

"id":<String_value>,

"secondary_server_type":<String_value>,

"primary_server_type":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/aaa_server/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aaa_server":[{

"external_servers":[{

"priority":<Integer_value>,

"parent_name":<String_value>,

"external_server_type":<String_value>,

"external_server_name":<String_value>,

"id":<String_value>,

"parent_id":<String_value>}],

"secondary_server_name":<String_value>,

"fallback_local_authentication":<Boolean_value>,

"log_ext_group_info":<Boolean_value>,

"enable_secondary_server":<Boolean_value>,

"primary_server_name":<String_value>,

"id":<String_value>,

"secondary_server_type":<String_value>,

"primary_server_type":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/aaa_server/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"aaa_server":{

<b>"primary_server_type":<String_value></b>,

"external_servers":[{

"priority":<Integer_value>,

"id":<String_value>,

"external_server_name":<String_value>,

"external_server_type":<String_value>}],

"id":<String_value>,

"fallback_local_authentication":<Boolean_value>,

"log_ext_group_info":<Boolean_value>,

"primary_server_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"aaa_server":[{

"external_servers":[{

"priority":<Integer_value>,

"parent_name":<String_value>,
```

```
"external_server_type":<String_value>,

"external_server_name":<String_value>,

"id":<String_value>,

"parent_id":<String_value>}],

"secondary_server_name":<String_value>,

"fallback_local_authentication":<Boolean_value>,

"log_ext_group_info":<Boolean_value>,

"enable_secondary_server":<Boolean_value>,

"primary_server_name":<String_value>,

"id":<String_value>,

"secondary_server_type":<String_value>,

"primary_server_type":<String_value>}]}
```

activity_status

Configuration for Activity Status resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the task logs.
status_code	<Integer>	Read-only	Error status code if any..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
status	<String>	Read-only	Status.
reboot_now	<Boolean>	Read-only	if true then the system is about to reboot.
activity	<String>	Read-only	Activity.
message	<String>	Read-only	Message.
act_id	<String>	Read-only	Activity Id.
is_last	<Boolean>	Read-only	is_last.
starttime	<Integer>	Read-only	Start Time.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/activity_status **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/activity_status **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of activity_status resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/activity_status **? pagesize=#no;pageno=#no** Use this query-parameter to get the activity_status resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/activity_status **?count=yes** Use this query-parameter to get the count of activity_status resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"activity_status":[{

"status_code":<Integer_value>,

"status":<String_value>,

"reboot_now":<Boolean_value>,

"activity":<String_value>,

"message":<String_value>,

"act_id":<String_value>,

"is_last":<Boolean_value>,

"starttime":<Integer_value>,

"id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/activity_status/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"activity_status":[{

"status_code":<Integer_value>,

"status":<String_value>,

"reboot_now":<Boolean_value>,

"activity":<String_value>,

"message":<String_value>,

"act_id":<String_value>,

"is_last":<Boolean_value>,

"starttime":<Integer_value>,

"id":<String_value>}]}
```

af_waf_deploy_rules_l2

Configuration for WAF Learning engine ready to deploy relaxation rules learnt for Level 2 resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
field_type	<String>	Read-write	Field type to be relaxed. Maximum length = 512
lastpolltime	<Double>	Read-write	Last Time polling was done to fetch hits.
status	<Boolean>	Read-write	Status is true when the rule is enabled and false when the rule is disabled.
mode	<Integer>	Read-write	Deployment mode would be OPEN(0) / CLOSE(1).
count_value	<Integer>	Read-write	count for the number of times this rule has caused violation.
hits	<Integer>	Read-write	Number of hits for the relaxation rule in this appfw profile..
failure_reason	<String>	Read-write	Field used to help the user figure out point of failure for a rule.
ctnsappname	<String>	Read-write	ctnsappname. Maximum length = 255
violation_location	<Integer>	Read-write	Violation Location.
violation_id	<Integer>	Read-write	Violation Category.
field_name	<String>	Read-write	Field name to be relaxed.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 512
resourceid	<String>	Read-write	Resourceid generated for each relaxation rule.. Maximum length = 128
is_system	<Boolean>	Read-write	Set to true when system generated and false for user configured rule.
max_value	<Integer>	Read-write	Maximum length of the filed to be relaxed.
rpt_sample_time	<Double>	Read-write	Report Sample time..
user_name	<String>	Read-write	Field used to maintain the name of User who take an action that lead to this transaction record.
learning_mode	<Integer>	Read-write	Learning mode to differentate between a rule to be pruned (2) or deployed (0 or 1).
value_type	<String>	Read-write	Value type to be relaxed. Maximum length = 512
lasthittime	<Double>	Read-write	Last hit for the relaxation rule in this appfw profile..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
value_expr	<String>	Read-write	Value expression to be relaxed. Maximum length = 512
http_req_url	<String>	Read-write	HTTP Request URL to be relaxed. Maximum length = 2000
is_deployed	<Integer>	Read-write	Field refelecting status of rule i.e 0-ready to deploy,1-deployed,2-skipped,4-failed.
si_device_ip_address	<String>	Read-write	Citrix ADC IP Address.. Minimum length = 1 Maximum length = 512
min_value	<Integer>	Read-write	minimum length of the field to be relaxed.
profile_name	<String>	Read-write	profile_name. Maximum length = 32
adm_learn_profile_name	<String>	Read-write	adm_learn_profile_name. Minimum length = 1 Maximum length = 32
id	<String>	Read-only	Id is system generated key..

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/af_waf_deploy_rules_l2 **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/af_waf_deploy_rules_l2 **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of af_waf_deploy_rules_l2 resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/af_waf_deploy_rules_l2 **?pagesize=#no;pageno=#no** Use this query-parameter to get the af_waf_deploy_rules_l2 resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/af_waf_deploy_rules_l2 **?count=yes** Use this query-parameter to get the count of af_waf_deploy_rules_l2 resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"af_waf_deploy_rules_l2":[{

"field_type":<String_value>,

"lastpolltime":<Double_value>,

"status":<Boolean_value>,

"mode":<Integer_value>,
```

```
"count_value":<Integer_value>,

"hits":<Integer_value>,

"failure_reason":<String_value>,

"ctnsappname":<String_value>,

"violation_location":<Integer_value>,

"id":<String_value>,

"violation_id":<Integer_value>,

"field_name":<String_value>,

"resourceid":<String_value>,

"is_system":<Boolean_value>,

"max_value":<Integer_value>,

"rpt_sample_time":<Double_value>,

"user_name":<String_value>,

"learning_mode":<Integer_value>,

"value_type":<String_value>,

"lasthittime":<Double_value>,

"value_expr":<String_value>,

"http_req_url":<String_value>,

"is_deployed":<Integer_value>,

"si_device_ip_address":<String_value>,

"min_value":<Integer_value>,

"profile_name":<String_value>,
```

```
"adm_learn_profile_name":<String_value>}]}
```

allowed_clients

Configuration for Filter Traffic in Management Service resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
redirect_ip	<String>	Read-write	Citrix SD-WAN IP to redirect.
id	<String>	Read-write	Id is system generated key.
accept_ips	<String[]>	Read-write	List of IPs to allow for NITRO requests.

Operations

(click to see Properties)

- ADD
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/allowed_clients?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"allowed_clients": {

"redirect_ip":<String_value>,

"id":<String_value>,
```

```
"accept_ips":<String_value[]>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"allowed_clients":[{

"redirect_ip":<String_value>,

"accept_ips_db":<String_value>,

"id":<String_value>,

"accept_ips":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/allowed_clients **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"allowed_clients":[{

"redirect_ip":<String_value>,

"accept_ips_db":<String_value>,

"id":<String_value>,

"accept_ips":<String_value>}]}
```

audit_log

Configuration for Audit log messages resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the audit log entries.
command	<String>	Read-write	Command given for this action.
status	<String>	Read-only	Status of this action whether accepted/ rejected by system.
port	<Integer>	Read-only	Client port from where operation specified for this entry was performed.
message	<String>	Read-only	Message for this action.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
username	<String>	Read-only	User Name, who performed operation specified for this entry.
client_type	<String>	Read-only	Client Type.
audittime	<Integer>	Read-only	Action Time on which operation specified for this entry was started.
ip_address	<String>	Read-only	Client IP Address from where operation specified for this entry was performed.
resource_name	<String>	Read-only	Resource Name on which operation specified for this entry was performed.
operation	<String>	Read-only	Operation Type that is performed.
resource_type	<String>	Read-only	Resource Type on which operation specified for this entry was performed .

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/audit_log **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/audit_log **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of audit_log resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/audit_log **? pagesize=#no;pageno=#no** Use this query-parameter to get the audit_log resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/audit_log **?count=yes** Use this query-parameter to get the count of audit_log resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"audit_log":[{

"status":<String_value>,

"port":<Integer_value>,

"message":<String_value>,

"username":<String_value>,

"client_type":<String_value>,

"tenant_id":<String_value>,

"audittime":<Integer_value>,

"id":<String_value>,
```

```
"ip_address":<String_value>,

"resource_name":<String_value>,

"operation":<String_value>,

"command":<String_value>,

"resource_type":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/audit_log/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"audit_log":[{

"status":<String_value>,

"port":<Integer_value>,

"message":<String_value>,
```

```
"username":<String_value>,

"client_type":<String_value>,

"tenant_id":<String_value>,

"audittime":<Integer_value>,

"id":<String_value>,

"ip_address":<String_value>,

"resource_name":<String_value>,

"operation":<String_value>,

"command":<String_value>,

"resource_type":<String_value>}]}
```

autoselection_preference

Configuration for Holds information regarding VIP and thirdparty VIP autoselection preference. resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
enable_non_access_vserver	<Boolean>	Read-write	Enable auto selection of non Accesible Vservers.
enable_auto_tp_vserver_selection	<Boolean>	Read-write	Enable auto selection of Third Party Vservers.
enable_auto_vip_selection	<Boolean>	Read-write	Enable auto selection of vips.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
type	<String>	Read-write	.
id	<String>	Read-write	Primary key for this table..

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/autoselection_preference **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/autoselection_preference **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of autoselection_preference resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/autoselection_preference **?pagesize=#no;pageno=#no** Use this query-parameter to get the autoselection_preference resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/autoselection_preference **?count=yes** Use this query-parameter to get the count of autoselection_preference resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"autoselection_preference":[{

"enable_non_access_vserver":<Boolean_value>,

"enable_auto_tp_vserver_selection":<Boolean_value>,

"enable_auto_vip_selection":<Boolean_value>,

"type":<String_value>,

"id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/autoselection_preference/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"autoselection_preference":[{

"enable_non_access_vserver":<Boolean_value>,
```

```
"enable_auto_tp_vserver_selection":<Boolean_value>,

"enable_auto_vip_selection":<Boolean_value>,

"type":<String_value>,

"id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/autoselection_preference/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"autoselection_preference":{

"id":<String_value>,

"enable_non_access_vserver":<Boolean_value>,

"enable_auto_tp_vserver_selection":<Boolean_value>,

"type":<String_value>,

"enable_auto_vip_selection":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"autoselection_preference":[{

"enable_non_access_vserver":<Boolean_value>,

"enable_auto_tp_vserver_selection":<Boolean_value>,

"enable_auto_vip_selection":<Boolean_value>,

"type":<String_value>,

"id":<String_value>}]}
```

backup

Configuration for backup resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
backup_file_name	<String>	Read-write	Backup file name. Maximum length = 64
backup_password	<String>	Read-write	Password of encrypted backup file.

Operations

(click to see Properties)

- ADD
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/backup?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"backup": {

"backup_file_name":<String_value>,

"backup_password":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"backup":[{

"backup_file_name":<String_value>,

"backup_password":<String_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/backup **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/backup **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of backup resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/backup **? pagesize=#no;pageno=#no** Use this query-parameter to get the backup resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/backup **?count=yes** Use this query-parameter to get the count of backup resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"backup":[{

"backup_file_name":<String_value>,

"backup_password":<String_value>}]}
```

backup_component

Configuration for backup components resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
component_message	<String>	Read-write	Information about the componenet.
component_versions	<String>	Read-write	Backup component version.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
component_minimum_version	<String>	Read-write	Minimum version supported by the Citrix ADC.
backup_file_name	<String>	Read-write	Backup file name. Maximum length = 64
component	<String>	Read-write	Backup component.
component_name	<String>	Read-write	Backup component Name.
component_ip	<String>	Read-write	Backup component IPaddress.
component_version_bit	<String>	Read-write	(32/64 bit) information for backup component version.
component_hostname	<String>	Read-write	Backup component VM Name.
component_error_severity	<Integer>	Read-write	Severity of error for the component.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

backup_external_storage

Configuration for Backup File External storage resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_name	<String>	Read-write	.
delete_from_svm	<Boolean>	Read-write	Password for backup file encryption.
port	<Integer>	Read-write	External transfer server port. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_id	<String>	Read-write	.
username	<String>	Read-write	Username.
transfer_protocol	<String>	Read-write	File transfer Protocol: SCP, SFTP or FTP.
password	<String>	Read-write	Password.
directory_path	<String>	Read-write	External server directory path where backup file will be uploaded.
id	<String>	Read-write	Id is system generated key for all backed up files.
server	<String>	Read-write	External server IP Address/Hostname.

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/backup_external_storage **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/backup_external_storage **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of backup_external_storage resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/backup_external_storage **?pagesize=#no;pageno=#no** Use this query-parameter to get the backup_external_storage resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/backup_external_storage **?count=yes** Use this query-parameter to get the count of backup_external_storage resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"backup_external_storage":[{

"parent_name":<String_value>,

"delete_from_svm":<Boolean_value>,

"port":<Integer_value>,

"parent_id":<String_value>,

"username":<String_value>,

"transfer_protocol":<String_value>,

"password":<String_value>,

"directory_path":<String_value>,

"id":<String_value>,

"server":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/backup_external_storage/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"backup_external_storage":[{

"parent_name":<String_value>,

"delete_from_svm":<Boolean_value>,

"port":<Integer_value>,

"parent_id":<String_value>,

"username":<String_value>,

"transfer_protocol":<String_value>,

"password":<String_value>,

"directory_path":<String_value>,

"id":<String_value>,
```

```
"server":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/backup_external_storage/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"backup_external_storage":{

"parent_id":<String_value>,

"password":<String_value>,

"server":<String_value>,

"id":<String_value>,

"parent_name":<String_value>,

"port":<Integer_value>,

"username":<String_value>,

"delete_from_svm":<Boolean_value>,

"transfer_protocol":<String_value>,

"directory_path":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"backup_external_storage":[{

"parent_name":<String_value>,

"delete_from_svm":<Boolean_value>,

"port":<Integer_value>,

"parent_id":<String_value>,

"username":<String_value>,

"transfer_protocol":<String_value>,

"password":<String_value>,

"directory_path":<String_value>,

"id":<String_value>,

"server":<String_value>}]}
```

backup_file

Configuration for Backup file of system resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
encrypted	<Boolean>	Read-write	Encrypted backup files.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
external_storage_info	<backup_external_storage>	Read-write	Information of the External storage for backup file.
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Double>	Read-only	File Size.
file_last_modified	<Double>	Read-only	Last Modified.

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/backup_file/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/backup_file **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/backup_file **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of backup_file resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/backup_file **? pagesize=#no;pageno=#no** Use this query-parameter to get the backup_file resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/backup_file **?count=yes** Use this query-parameter to get the count of backup_file resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"backup_file":[{

"encrypted":<Boolean_value>,
```

```
"file_location_path":<String_value>,

"file_size":<Double_value>,

"external_storage_info":<backup_external_storage_value>,

"file_name":<String_value>,

"file_last_modified":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/backup_file/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"backup_file":[{

"encrypted":<Boolean_value>,

"file_location_path":<String_value>,
```

```
"file_size":<Double_value>,

"external_storage_info":<backup_external_storage_value>,

"file_name":<String_value>,

"file_last_modified":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/backup_file **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

backup_policy

Configuration for Backup Policy resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
backup_to_retain	<Integer>	Read-write	Number of previous backups to retain. Minimum value = 1 Maximum value =
enable_external_transfer	<Boolean>	Read-write	Enable transfer of backup file to external server.
policy_name	<String>	Read-write	Policy Name. Minimum length = 1 Maximum length = 128
external_storage_info	<backup_external_storage>	Read-write	Information of the External storage for backup file.
backup_password	<String>	Read-write	Password for backup file encryption.
encrypt_backup_file	<Boolean>	Read-write	Encrypts backup files.
id	<String>	Read-write	Id is system generated key.

Operations

(click to see Properties)

- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get

URL: https://<MGMT-IP>/nitro/v2/config/backup_policy **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"backup_policy":[{

"backup_to_retain":<Integer_value>,

"enable_external_transfer":<Boolean_value>,

"policy_name":<String_value>,

"external_storage_info":<backup_external_storage_value>,

"backup_password":<String_value>,

"encrypt_backup_file":<Boolean_value>,

"id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/backup_policy/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"backup_policy":{

<b>"backup_to_retain":<Integer_value></b>,

"policy_name":<String_value>,

"external_storage_info":<backup_external_storage_value>,

"id":<String_value>,

"encrypt_backup_file":<Boolean_value>,

"enable_external_transfer":<Boolean_value>,

"backup_password":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"backup_policy":[{

"backup_to_retain":<Integer_value>,

"enable_external_transfer":<Boolean_value>,

"policy_name":<String_value>,

"external_storage_info":<backup_external_storage_value>,

"backup_password":<String_value>,

"encrypt_backup_file":<Boolean_value>,

"id":<String_value>}]}
```

built_in_agent

Configuration for Built-in Agent resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Built-in Agent Name (IP Address).
id	<String>	Read-write	Built-in Agent ID.
mastools_ws_state	<String>	Read-only	WebSocket State of built in agent, UP only if Web Socket is connected.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/built_in_agent?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"built_in_agent": {

"id":<String_value>,

"name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"built_in_agent":[{

"mastools_ws_state":<String_value>,

"name":<String_value>,

"id":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/built_in_agent/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/built_in_agent **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/built_in_agent **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of built_in_agent resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/built_in_agent **?pagesize=#no;pageno=#no** Use this query-parameter to get the built_in_agent resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/built_in_agent **?count=yes** Use this query-parameter to get the count of built_in_agent resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"built_in_agent":[{

"mastools_ws_state":<String_value>,

"name":<String_value>,

"id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/built_in_agent/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"built_in_agent":[{

"mastools_ws_state":<String_value>,

"name":<String_value>,

"id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/built_in_agent/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"built_in_agent":{

<b>"id":<String_value></b>,

"name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"built_in_agent":[{

"mastools_ws_state":<String_value>,

"name":<String_value>,

"id":<String_value>}]}
```

capacity

Configuration for License pool capacity of appliance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
std_bw	<Integer>	Read-write	Standard Bandwidth pool in Gbps.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ent_bw	<Integer>	Read-write	Enterprise Bandwidth pool in Gbps.
plt_bw	<Integer>	Read-write	Platinum Bandwidth pool in Gbps.
id	<String>	Read-write	Id is system generated key.
instance	<Integer>	Read-write	Instance bandwidth pool.

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/capacity **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/capacity **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of capacity resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/capacity **? pagesize=#no;pageno=#no** Use this query-parameter to get the capacity resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/capacity **?count=yes** Use this query-parameter to get the count of capacity resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"capacity":[{

"std_bw":<Integer_value>,

"ent_bw":<Integer_value>,

"plt_bw":<Integer_value>,

"id":<String_value>,

"instance":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/capacity/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"capacity":[{

"std_bw":<Integer_value>,

"ent_bw":<Integer_value>,

"plt_bw":<Integer_value>,

"id":<String_value>,

"instance":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/capacity/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"capacity":{

"std_bw":<Integer_value>,

"ent_bw":<Integer_value>,

"id":<String_value>,

"instance":<Integer_value>,

"plt_bw":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"capacity":[{

"std_bw":<Integer_value>,

"ent_bw":<Integer_value>,

"plt_bw":<Integer_value>,

"id":<String_value>,

"instance":<Integer_value>}]}
```

cb_sb_image

Configuration for CBSDX Single Bundle Build File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
is_dir	<Boolean>	Read-write	Is the file a directory?.
file_size	<String>	Read-only	file_size.
modified_timestamp	<String>	Read-only	Last Modified.

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/cb_sb_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/cb_sb_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/cb_sb_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of cb_sb_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/cb_sb_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the cb_sb_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/cb_sb_image **?count=yes** Use this query-parameter to get the count of cb_sb_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cb_sb_image":[{

"file_size":<String_value>,

"modified_timestamp":<String_value>,

"file_name":<String_value>,

"is_dir":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/cb_sb_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cb_sb_image":[{

"file_size":<String_value>,

"modified_timestamp":<String_value>,

"file_name":<String_value>,

"is_dir":<Boolean_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/cb_sb_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

cb_sb_upgrade

Configuration for CBSDX Single Bundle Upgrade resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
ns_ip_arr_to_save	<String[]>	Read-write	List of Citrix ADC IP address for which config needs to be saved.

Operations

(click to see Properties)

- UPGRADE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

upgrade

URL: https://<MGMT-IP>/nitro/v2/config/cb_sb_upgrade?action=upgrade;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"cb_sb_upgrade": {

"ns_ip_arr_to_save":<String_value[]>,
```

```
"file_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cb_sb_upgrade":[{

"file_name":<String_value>,

"ns_ip_arr_to_save":<String_value>}]}
```

cb_sb_upgrade_monitor

Configuration for CBSDX Single Bundle Upgrade Status resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
component_name	<String>	Read-write	Component Name. Minimum length = 1 Maximum length = 256
time_remaining	<String>	Read-only	Remaining Time.
reboot_required	<String>	Read-only	Reboot Required.
status_message	<String>	Read-only	Upgrade Status Message.
error_code	<String>	Read-only	Error Code.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/cb_sb_upgrade_monitor **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/cb_sb_upgrade_monitor **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of cb_sb_upgrade_monitor resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/cb_sb_upgrade_monitor **?pagesize=#no;pageno=#no** Use this query-parameter to get the cb_sb_upgrade_monitor resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/cb_sb_upgrade_monitor **?count=yes** Use this query-parameter to get the count of cb_sb_upgrade_monitor resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cb_sb_upgrade_monitor":[{

"time_remaining":<String_value>,

"component_name":<String_value>,

"reboot_required":<String_value>,

"status_message":<String_value>,
```

```
"error_code":<String_value>}]}
```

central_management_server

Configuration for Central managment server resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
password	<String>	Read-write	Password of central management server. Minimum length = 1 Maximum length = 128
server	<String>	Read-write	Server/IP address of central management server. Minimum length = 1 Maximum length = 255
type	<String>	Read-write	Type of central management server. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key.
user_name	<String>	Read-write	User Name of central management server. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128

Operations

(click to see Properties)

- ADD
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/central_management_server?onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"central_management_server": {

<b>"server":<String_value></b>,

"password":<String_value>,

"id":<String_value>,

"user_name":<String_value>,

"type":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"central_management_server":[{

"password":<String_value>,

"server":<String_value>,

"type":<String_value>,

"id":<String_value>,

"user_name":<String_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/central_management_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/central_management_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of central_management_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/central_management_server **?pagesize=#no;pageno=#no** Use this query-parameter to get the central_management_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/central_management_server **?count=yes** Use this query-parameter to get the count of central_management_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"central_management_server":[{

"password":<String_value>,

"server":<String_value>,

"type":<String_value>,

"id":<String_value>,

"user_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/central_management_server/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"central_management_server":[{

"password":<String_value>,

"server":<String_value>,

"type":<String_value>,

"id":<String_value>,

"user_name":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/central_management_server/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"central_management_server":{

<b>"server":<String_value></b>,

"password":<String_value>,

"id":<String_value>,

"user_name":<String_value>,
```

```
"type":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"central_management_server":[{

"password":<String_value>,

"server":<String_value>,

"type":<String_value>,

"id":<String_value>,

"user_name":<String_value>}]}
```

cipher_config

Configuration for SSL Cipher Config resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cipher_group_name	<String>	Read-write	Name of the Cipher Group. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
config_mode	<String>	Read-write	SSL Ciphers Config Mode [CipherGroup, CipherSuites].
cipher_name_list_array	<String[]>	Read-write	list of cipher suites in form of array of strings.

Operations

(click to see Properties)

- GET (ALL)
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/cipher_config **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/cipher_config **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of cipher_config resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/cipher_config **? pagesize=#no;pageno=#no** Use this query-parameter to get the cipher_config resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/cipher_config **?count=yes** Use this query-parameter to get the count of cipher_config resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cipher_config":[{

"cipher_group_name":<String_value>,

"config_mode":<String_value>,

"cipher_name_list_array":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/cipher_config/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"cipher_config":{

"config_mode":<String_value>,

"cipher_group_name":<String_value>,

"cipher_name_list_array":<String_value[]>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cipher_config":[{

"cipher_group_name":<String_value>,

"config_mode":<String_value>,

"cipher_name_list_array":<String_value>}]}
```

cipher_group

Configuration for Cipher Group resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cipher_group_description	<String>	Read-write	Describing the Cipher Group algorithms created. Minimum length = 1 Maximum length = 256
cipher_group_name	<String>	Read-write	Name of Cipher Group. Minimum length = 1 Maximum length = 128
cipher_name_list_array	<String[]>	Read-write	list of cipher suites in form of array of strings.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/cipher_group?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"cipher_group": {

<b>"cipher_group_description":<String_value></b>,

<b>"cipher_group_name":<String_value></b>,

<b>"cipher_name_list_array":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cipher_group":[{

"cipher_group_description":<String_value>,

"is_applied":<Boolean_value>,

"cipher_group_name":<String_value>,

"is_predefined":<Boolean_value>,

"cipher_name_list_str":<String_value>,
```

```
"cipher_name_list_array":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/cipher_group/cipher_group_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/cipher_group **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/cipher_group **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of cipher_group resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/cipher_group **? pagesize=#no;pageno=#no** Use this query-parameter to get the cipher_group resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/cipher_group **?count=yes** Use this query-parameter to get the count of cipher_group resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cipher_group":[{

"cipher_group_description":<String_value>,

"is_applied":<Boolean_value>,

"cipher_group_name":<String_value>,

"is_predefined":<Boolean_value>,

"cipher_name_list_str":<String_value>,

"cipher_name_list_array":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/cipher_group/cipher_group_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cipher_group":[{

"cipher_group_description":<String_value>,

"is_applied":<Boolean_value>,

"cipher_group_name":<String_value>,

"is_predefined":<Boolean_value>,

"cipher_name_list_str":<String_value>,

"cipher_name_list_array":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/cipher_group/cipher_group_name_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"cipher_group":{

"cipher_group_name":<String_value>,

"cipher_group_description":<String_value>,

"cipher_name_list_array":<String_value[]>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cipher_group":[{

"cipher_group_description":<String_value>,

"is_applied":<Boolean_value>,

"cipher_group_name":<String_value>,

"is_predefined":<Boolean_value>,

"cipher_name_list_str":<String_value>,

"cipher_name_list_array":<String_value>}]}
```

cipher_suite

Configuration for Cipher Suite resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cipher_suite_name	<String>	Read-write	Name of the Cipher Suite. Minimum length = 1 Maximum length = 128

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/cipher_suite **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/cipher_suite **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of cipher_suite resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/cipher_suite **? pagesize=#no;pageno=#no** Use this query-parameter to get the cipher_suite resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/cipher_suite **?count=yes** Use this query-parameter to get the count of cipher_suite resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cipher_suite":[{

"protocol":<String_value>,

"message_auth_code":<String_value>,

"encryption":<String_value>,

"cipher_suite_openssl_name":<String_value>,

"authentication":<String_value>,

"cipher_suite_description":<String_value>,

"is_supported":<Boolean_value>,

"preference":<Integer_value>,

"cipher_suite_name":<String_value>,

"hexcode":<String_value>,

"key_exchange":<String_value>,

"cipher_group_list":<String_value>}]}
```

clean_install_info

Configuration for Single Bundle File Info For Clean Install resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_name	<String>	Read-write	File Name. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 256
total_approx_time	<Integer>	Read-only	Total approximate time that the upgrade will take.
component_info	<String[]>	Read-only	Component Info.
selected_sb_version	<String>	Read-only	Selected Single Bundle Version.
error_message	<String>	Read-only	Error Message in case the Single Bundle cannot be applied.
reboot_message	<String>	Read-only	Reboot Message.
current_sb_version	<String>	Read-only	Current Single Bundle Version.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

cli_prompt

Configuration for CLI Prompt string resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
promptString	<String>	Read-write	The prompt string. The following special values are allowed: %! - will be replaced by the history event number %u - will be replaced by the Citrix ADC SVM user name %h - will be replaced by the Citrix ADC SVM hostname %t - will be replaced by the current time (12 hr) %T - will be

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			replaced by the current time (24 hr) %d - will be replaced by the current date (yy/mm/dd) This is a mandatory argument..

Operations

(click to see Properties)

- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

update

URL: https://<MGMT-IP>/nitro/v2/config/cli_prompt/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"cli_prompt":{

<b>"promptString":<String_value></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cli_prompt":[{

"promptString":<String_value>}]}
```

counter_item

Configuration for Counter items resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
timestamp	<Double>	Read-write	Timestamp of the anomalous counter.
value	<Double>	Read-write	Value of the anomalous counter.
anomaly	<Boolean>	Read-write	If the counter item is anomalous.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

csv_export

Configuration for CSV Export resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
report_end_time	<Double>	Read-write	Report end time..
order_by	<String>	Read-write	Order By. Maximum length = 32
args	<String>	Read-write	args.
pageno	<Integer>	Read-write	Page no..
report_start_time	<Double>	Read-write	Report start time..
resource_id	<String>	Read-write	Resource id. Maximum length = 512
asc	<String>	Read-write	asc. Maximum length = 32
duration_summary	<String>	Read-write	Duration summary.. Maximum length = 16

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is CSV export Id. Maximum length = 32
section_title	<String>	Read-write	Section title. Maximum length = 512
attrs	<String>	Read-write	attrs.
is_count	<String>	Read-write	is_count. Maximum length = 32
report_type	<String>	Read-write	report_type 'active' in case active report.
filter	<String>	Read-write	filter.
detailview	<String>	Read-write	detailview. Maximum length = 64
duration	<String>	Read-write	Duration. Maximum length = 64
graphql_query	<String>	Read-write	GraphQL query for export.
cr_enabled	<String>	Read-write	CR Enabled.. Maximum length = 16
sla_enabled	<String>	Read-write	SLA Enabled.. Maximum length = 16
type	<String>	Read-write	Type.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 32
resource_name	<String>	Read-write	Resource name. Minimum length = 1 Maximum length = 255
pagesize	<Integer>	Read-write	Page size.
csv_mapping_id	<String>	Read-write	Schedule export csv_mapping_id. Maximum length = 32

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/csv_export?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"csv_export": {

"pageno":<Integer_value>,

"duration_summary":<String_value>,

"id":<String_value>,
```

```
"section_title":<String_value>,

"attrs":<String_value>,

"report_type":<String_value>,

"detailview":<String_value>,

"duration":<String_value>,

"cr_enabled":<String_value>,

"sla_enabled":<String_value>,

"resource_name":<String_value>,

"type":<String_value>,

"csv_mapping_id":<String_value>,

"report_end_time":<Double_value>,

"args":<String_value>,

"order_by":<String_value>,

"resource_id":<String_value>,

"report_start_time":<Double_value>,

"asc":<String_value>,

"is_count":<String_value>,

"filter":<String_value>,

"graphql_query":<String_value>,

"pagesize":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"csv_export":[{

"report_end_time":<Double_value>,

"order_by":<String_value>,

"args":<String_value>,

"pageno":<Integer_value>,

"report_start_time":<Double_value>,

"resource_id":<String_value>,

"asc":<String_value>,

"duration_summary":<String_value>,

"id":<String_value>,

"section_title":<String_value>,

"attrs":<String_value>,

"is_count":<String_value>,

"report_type":<String_value>,

"filter":<String_value>,

"detailview":<String_value>,

"duration":<String_value>,

"graphql_query":<String_value>,
```

```
"cr_enabled":<String_value>,

"sla_enabled":<String_value>,

"type":<String_value>,

"resource_name":<String_value>,

"pagesize":<Integer_value>,

"csv_mapping_id":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/csv_export/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/csv_export **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/csv_export **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of csv_export resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/csv_export **?pagesize=#no;pageno=#no** Use this query-parameter to get the csv_export resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/csv_export **?count=yes** Use this query-parameter to get the count of csv_export resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"csv_export":[{

"report_end_time":<Double_value>,

"order_by":<String_value>,

"args":<String_value>,

"pageno":<Integer_value>,

"report_start_time":<Double_value>,

"resource_id":<String_value>,

"asc":<String_value>,
```

```
"duration_summary":<String_value>,

"id":<String_value>,

"section_title":<String_value>,

"attrs":<String_value>,

"is_count":<String_value>,

"report_type":<String_value>,

"filter":<String_value>,

"detailview":<String_value>,

"duration":<String_value>,

"graphql_query":<String_value>,

"cr_enabled":<String_value>,

"sla_enabled":<String_value>,

"type":<String_value>,

"resource_name":<String_value>,

"pagesize":<Integer_value>,

"csv_mapping_id":<String_value>}]}
```

current_hostname

Configuration for System Hostname resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hostname	<String>	Read-write	Hostname. Minimum length = 1 Maximum length = 63

Operations

(click to see Properties)

- GET (ALL)
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/current_hostname **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/current_hostname **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of current_hostname resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/current_hostname **?pagesize=#no;pageno=#no** Use this query-parameter to get the current_hostname resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/current_hostname **?count=yes** Use this query-parameter to get the count of current_hostname resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"current_hostname":[{

"hostname":<String_value>}]}
```

modify

URL: https://<MGMT-IP>/nitro/v2/config/current_hostname/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"current_hostname":{

"hostname":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"current_hostname":[{

"hostname":<String_value>}]}
```

current_timezone

Configuration for Current timezone resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
timezone	<String>	Read-write	Timezone.

Operations

(click to see Properties)

- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get

URL: https://<MGMT-IP>/nitro/v2/config/current_timezone **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"current_timezone":[{

"timezone":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/current_timezone/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"current_timezone":{

"timezone":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"current_timezone":[{

"timezone":<String_value>}]}
```

device_backup

Configuration for Device Backup resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
wallet_key_encrypted	<Boolean>	Read-write	True if wallet key encrypted for service deployment.
last_updated_time	<Double>	Read-write	Last Updated Time.
backup_filesize	<Double>	Read-write	Backup file size.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all backed up files.
ip_address	<String>	Read-write	Device IP Address. Minimum length = 1 Maximum length = 64
encrypted	<Boolean>	Read-write	Indicates if the backup files are encrypted.
backup_password	<String>	Read-write	Backup Password.
use_global_password	<Boolean>	Read-write	Global Password.
device_name	<String>	Read-only	Device Hostname.
backup_filename	<String>	Read-only	Name of the backed up file.

Operations

(click to see Properties)

- DEVICEBACKUP
- UPLOAD
- DELETE
- GET (ALL)
- DEVICERESTORE
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

devicebackup

URL: https://<MGMT-IP>/nitro/v2/config/device_backup?action=devicebackup;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"device_backup": {

"last_updated_time":<Double_value>,

"id":<String_value>,

"wallet_key_encrypted":<Boolean_value>,

"backup_filesize":<Double_value>,

"backup_password":<String_value>,

"use_global_password":<Boolean_value>,

"ip_address":<String_value>,

"encrypted":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_backup":[{

"external_storage_info":<backup_external_storage_value>,

"device_family":<String_value>,

"wallet_key_encrypted":<Boolean_value>,

"image_filesize":<Double_value>,

"last_updated_time":<Double_value>,
```

```
"backup_filesize":<Double_value>,

"tenant_id":<String_value>,

"device_name":<String_value>,

"image_filename":<String_value>,

"backup_filename":<String_value>,

"id":<String_value>,

"ip_address":<String_value>,

"encrypted":<Boolean_value>,

"act_id":<String_value>,

"backup_password":<String_value>,

"use_global_password":<Boolean_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/device_backup/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/device_backup **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/device_backup **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of device_backup resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/device_backup **? pagesize=#no;pageno=#no** Use this query-parameter to get the device_backup resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/device_backup **?count=yes** Use this query-parameter to get the count of device_backup resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_backup":[{

"external_storage_info":<backup_external_storage_value>,

"device_family":<String_value>,

"wallet_key_encrypted":<Boolean_value>,

"image_filesize":<Double_value>,

"last_updated_time":<Double_value>,

"backup_filesize":<Double_value>,

"tenant_id":<String_value>,

"device_name":<String_value>,

"image_filename":<String_value>,

"backup_filename":<String_value>,

"id":<String_value>,

"ip_address":<String_value>,

"encrypted":<Boolean_value>,

"act_id":<String_value>,

"backup_password":<String_value>,

"use_global_password":<Boolean_value>}]}
```

devicerestore

URL: https://<MGMT-IP>/nitro/v2/config/device_backup?action=devicerestore;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"device_backup": {

"last_updated_time":<Double_value>,

"id":<String_value>,

"wallet_key_encrypted":<Boolean_value>,

"backup_filesize":<Double_value>,

"backup_password":<String_value>,

"use_global_password":<Boolean_value>,

"ip_address":<String_value>,

"encrypted":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_backup":[{

"external_storage_info":<backup_external_storage_value>,

"device_family":<String_value>,

"wallet_key_encrypted":<Boolean_value>,
```

```
"image_filesize":<Double_value>,

"last_updated_time":<Double_value>,

"backup_filesize":<Double_value>,

"tenant_id":<String_value>,

"device_name":<String_value>,

"image_filename":<String_value>,

"backup_filename":<String_value>,

"id":<String_value>,

"ip_address":<String_value>,

"encrypted":<Boolean_value>,

"act_id":<String_value>,

"backup_password":<String_value>,

"use_global_password":<Boolean_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/device_backup **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

device_backup_policy

Configuration for Device Backup Policy resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
backup_on_cfg_sav_trap	<Boolean>	Read-write	Backup on recieving NetscalerConfigSave trap.
backup_ontrap_delay	<Integer>	Read-write	Delay in taking Backup on Receiving NetscalerConfigSave trap.
sdx_backup_timeout	<Integer>	Read-write	Backup Timeout for SDX Backup.
encrypt_backup_file	<Boolean>	Read-write	Encrypts backup files.
enable_scheduled_backups	<Boolean>	Read-write	Enable Scheduled Backup (default is enabled).

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
enable_external_transfer	<Boolean>	Read-write	Enable transfer of device backup files to external server.
polling_interval	<Integer>	Read-write	Frequency of device backup in hours.
backup_times	<String>	Read-write	Comma Separated string of backup times to be scheduled..
number_of_backups	<Integer>	Read-write	Number of backup files maintained per device. Minimum value = 1
is_interval_based	<Boolean>	Read-write	Parameter to decide whether backup setting is interval based or time based.
enable_geodb_backups	<Boolean>	Read-write	Enable GeoDB Backup from Citrix ADC (default is disabled).

Operations

(click to see Properties)

- DO_POLL
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

do_poll

URL: https://<MGMT-IP>/nitro/v2/config/device_backup_policy?action=do_poll;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"device_backup_policy": {

<b>"number_of_backups":<Integer_value></b>,

<b>"is_interval_based":<Boolean_value></b>,

"enable_scheduled_backups":<Boolean_value>,

"encrypt_backup_file":<Boolean_value>,

"enable_external_transfer":<Boolean_value>,

"backup_times":<String_value>,

"backup_on_cfg_sav_trap":<Boolean_value>,

"backup_ontrap_delay":<Integer_value>,

"polling_interval":<Integer_value>,

"sdx_backup_timeout":<Integer_value>,

"enable_geodb_backups":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_backup_policy":[{

"external_storage_info":<backup_external_storage_value>,

"backup_on_cfg_sav_trap":<Boolean_value>,

"device_family":<String_value>,

"backup_ontrap_delay":<Integer_value>,
```

```
"backup_password":<String_value>,

"sdx_backup_timeout":<Integer_value>,

"encrypt_backup_file":<Boolean_value>,

"enable_scheduled_backups":<Boolean_value>,

"enable_external_transfer":<Boolean_value>,

"polling_interval":<Integer_value>,

"backup_times":<String_value>,

"number_of_backups":<Integer_value>,

"is_interval_based":<Boolean_value>,

"enable_geodb_backups":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/device_backup_policy **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_backup_policy":[{

"external_storage_info":<backup_external_storage_value>,

"backup_on_cfg_sav_trap":<Boolean_value>,

"device_family":<String_value>,

"backup_ontrap_delay":<Integer_value>,

"backup_password":<String_value>,

"sdx_backup_timeout":<Integer_value>,

"encrypt_backup_file":<Boolean_value>,

"enable_scheduled_backups":<Boolean_value>,

"enable_external_transfer":<Boolean_value>,

"polling_interval":<Integer_value>,

"backup_times":<String_value>,

"number_of_backups":<Integer_value>,

"is_interval_based":<Boolean_value>,

"enable_geodb_backups":<Boolean_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/device_backup_policy/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"device_backup_policy":{

<b>"number_of_backups":<Integer_value></b>,

<b>"is_interval_based":<Boolean_value></b>,

"enable_scheduled_backups":<Boolean_value>,

"encrypt_backup_file":<Boolean_value>,

"enable_external_transfer":<Boolean_value>,

"backup_times":<String_value>,

"backup_on_cfg_sav_trap":<Boolean_value>,

"backup_ontrap_delay":<Integer_value>,

"polling_interval":<Integer_value>,

"sdx_backup_timeout":<Integer_value>,

"enable_geodb_backups":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"device_backup_policy":[{

"external_storage_info":<backup_external_storage_value>,

"backup_on_cfg_sav_trap":<Boolean_value>,
```

```
"device_family":<String_value>,

"backup_ontrap_delay":<Integer_value>,

"backup_password":<String_value>,

"sdx_backup_timeout":<Integer_value>,

"encrypt_backup_file":<Boolean_value>,

"enable_scheduled_backups":<Boolean_value>,

"enable_external_transfer":<Boolean_value>,

"polling_interval":<Integer_value>,

"backup_times":<String_value>,

"number_of_backups":<Integer_value>,

"is_interval_based":<Boolean_value>,

"enable_geodb_backups":<Boolean_value>}]}
```

entity_map

Configuration for entity map resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
value	<String>	Read-write	value. Minimum length = 1 Maximum length = 1024
key	<String>	Read-write	key.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 1024

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

external_server

Configuration for External Server configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
priority	<Integer>	Read-write	Priority of external server. Minimum value = 2 Maximum value =
external_server_type	<String>	Read-write	Type of external server. Supported types 1.RADIUS 2.LDAP 3.TACACS 4.KEYSTONE. Minimum length = 1 Maximum length = 32
external_server_name	<String>	Read-write	Name of external server. Minimum length = 1 Maximum length = 128
id	<String>	Read-write	Id is system generated key for all the external servers.

Operations

(click to see Properties)

- GET (ALL)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/external_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/external_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of

external_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/external_server **?pagesize=#no;pageno=#no** Use this query-parameter to get the external_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/external_server **?count=yes** Use this query-parameter to get the count of external_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"external_server":[{

"priority":<Integer_value>,

"parent_name":<String_value>,

"external_server_type":<String_value>,

"external_server_name":<String_value>,

"id":<String_value>,
```

```
"parent_id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/external_server/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"external_server":[{

"priority":<Integer_value>,

"parent_name":<String_value>,

"external_server_type":<String_value>,

"external_server_name":<String_value>,

"id":<String_value>,

"parent_id":<String_value>}]}
```

jazz_license

Configuration for Jazz License Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
session_id	<String>	Read-write	Session id for license.
name	<String>	Read-write	Name of license.
serial_no	<String>	Read-write	Serial Number for license.
id	<String>	Read-write	ID for license.
bind_type	<String>	Read-write	Bind Type for license.
use_proxy	<Boolean>	Read-write	Use license proxy configured to connect to citrix backoffice.
date_purchased	<String>	Read-only	Purchase Date for license.
features	<String>	Read-only	Features for license.
relevance	<Integer>	Read-only	Relevance of license.
count_total	<Integer>	Read-only	Count Total of license.
count_available	<Integer>	Read-only	Count Available of license.
date_exp	<String>	Read-only	Expiry Date for license.

Operations

(click to see Properties)

- FETCH
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

fetch

URL: https://<MGMT-IP>/nitro/v2/config/jazz_license?action=fetch;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"jazz_license": {

"id":<String_value>,

"bind_type":<String_value>,

"use_proxy":<Boolean_value>,

"name":<String_value>,

"session_id":<String_value>,

"serial_no":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"jazz_license":[{

"date_purchased":<String_value>,

"session_id":<String_value>,
```

```
"features":<String_value>,

"relevance":<Integer_value>,

"name":<String_value>,

"serial_no":<String_value>,

"count_total":<Integer_value>,

"count_available":<Integer_value>,

"id":<String_value>,

"date_exp":<String_value>,

"bind_type":<String_value>,

"use_proxy":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/jazz_license **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"jazz_license":[{

"date_purchased":<String_value>,

"session_id":<String_value>,

"features":<String_value>,

"relevance":<Integer_value>,

"name":<String_value>,

"serial_no":<String_value>,

"count_total":<Integer_value>,

"count_available":<Integer_value>,

"id":<String_value>,

"date_exp":<String_value>,

"bind_type":<String_value>,

"use_proxy":<Boolean_value>}]}
```

ldap_server

Configuration for LDAP Server resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
auth_timeout	<Integer>	Read-write	The maximum number of seconds the system will wait

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			for a response from the LDAP server.
group_search_subattribute	<String>	Read-write	LDAP group search subattribute. Used to determine to which groups a group belongs.. Maximum length = 32
follow_referrals	<Boolean>	Read-write	Enable following LDAP referrals received from LDAP server.
search_filter	<String>	Read-write	The String to be combined with the default LDAP user search string to form the value. Maximum length = 256
group_attr_name	<String>	Read-write	The Attribute name for group extraction from the LDAP server. Maximum length = 32
authentication	<Boolean>	Read-write	Set to false for public key Authentication.
ldap_host_name	<String>	Read-write	Host Name on the certificate from LDAP Server. Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
change_password	<Boolean>	Read-write	Enable change of the user.
default_authentication_group	<String>	Read-write	This is the default group. Maximum length = 64
max_nesting_level	<Integer>	Read-write	Number of levels at which group extraction is allowed. Maximum value =
login_name	<String>	Read-write	The name attribute used by the system to query the external LDAP server. Maximum length = 32
id	<String>	Read-write	Id is system generated key for all the ldap servers.
ip_address	<String>	Read-write	The IP address of the LDAP server.. Minimum length = 1 Maximum length = 64
nested_group_extraction	<Boolean>	Read-write	Enable Nested Group Extraction.
group_search_attribute	<String>	Read-write	LDAP group search attribute. Used to determine to which groups a group belongs.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 32
ssh_public_key	<String>	Read-write	SSH public key attribute holds the public keys of the user. Maximum length = 64
group_name_identifier	<String>	Read-write	Name that uniquely identifies a group in LDAP server. Maximum length = 32
subattribute_name	<String>	Read-write	The Sub-Attribute name for group extraction from LDAP server. Maximum length = 32
name	<String>	Read-write	Name of LDAP server. Minimum length = 1 Maximum length = 128
validate_ldap_server_certs	<Boolean>	Read-write	Validate LDAP Server Certificate.
group_search_filter	<String>	Read-write	String to be combined with the default LDAP group search string to form the search value. Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
bind_dn	<String>	Read-write	The full distinguished name used to bind to the LDAP server. Maximum length = 128
port	<Integer>	Read-write	The port number on which the LDAP server is running. Maximum value =
base_dn	<String>	Read-write	The base or node where the ldapsearch should start. Maximum length = 128
sec_type	<String>	Read-write	The communication type between the system and the LDAP server.
max_ldap_referrals	<Integer>	Read-write	Maximum number of ldap referrals to follow.
type	<String>	Read-write	The type of LDAP server. Minimum length = 2 Maximum length = 64
bind_passwd	<String>	Read-write	The password used to bind to the LDAP server. Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
address_type	<Integer>	Read-only	Configuration Type. Values: 0: IPv4, 1: IPv6, -1: Hostname.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/ldap_server?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ldap_server": {

<b>"ip_address":<String_value></b>,

<b>"name":<String_value></b>,

<b>"type":<String_value></b>,

"auth_timeout":<Integer_value>,

"search_filter":<String_value>,

"ldap_host_name":<String_value>,

"authentication":<Boolean_value>,

"change_password":<Boolean_value>,
```

```
"default_authentication_group":<String_value>,

"id":<String_value>,

"login_name":<String_value>,

"group_name_identifier":<String_value>,

"bind_dn":<String_value>,

"port":<Integer_value>,

"sec_type":<String_value>,

"base_dn":<String_value>,

"bind_passwd":<String_value>,

"group_search_subattribute":<String_value>,

"follow_referrals":<Boolean_value>,

"group_attr_name":<String_value>,

"max_nesting_level":<Integer_value>,

"nested_group_extraction":<Boolean_value>,

"group_search_attribute":<String_value>,

"ssh_public_key":<String_value>,

"subattribute_name":<String_value>,

"validate_ldap_server_certs":<Boolean_value>,

"group_search_filter":<String_value>,

"max_ldap_referrals":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ldap_server":[{

"auth_timeout":<Integer_value>,

"group_search_subattribute":<String_value>,

"follow_referrals":<Boolean_value>,

"search_filter":<String_value>,

"group_attr_name":<String_value>,

"authentication":<Boolean_value>,

"ldap_host_name":<String_value>,

"change_password":<Boolean_value>,

"default_authentication_group":<String_value>,

"max_nesting_level":<Integer_value>,

"login_name":<String_value>,

"id":<String_value>,

"ip_address":<String_value>,

"nested_group_extraction":<Boolean_value>,

"group_search_attribute":<String_value>,

"ssh_public_key":<String_value>,

"group_name_identifier":<String_value>,
```

```
"subattribute_name":<String_value>,

"name":<String_value>,

"validate_ldap_server_certs":<Boolean_value>,

"group_search_filter":<String_value>,

"bind_dn":<String_value>,

"port":<Integer_value>,

"base_dn":<String_value>,

"sec_type":<String_value>,

"max_ldap_referrals":<Integer_value>,

"type":<String_value>,

"bind_passwd":<String_value>,

"address_type":<Integer_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/ldap_server/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ldap_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ldap_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ldap_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ldap_server **? pagesize=#no;pageno=#no** Use this query-parameter to get the ldap_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ldap_server **?count=yes** Use this query-parameter to get the count of ldap_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ldap_server":[{

"auth_timeout":<Integer_value>,
```

```
"group_search_subattribute":<String_value>,

"follow_referrals":<Boolean_value>,

"search_filter":<String_value>,

"group_attr_name":<String_value>,

"authentication":<Boolean_value>,

"ldap_host_name":<String_value>,

"change_password":<Boolean_value>,

"default_authentication_group":<String_value>,

"max_nesting_level":<Integer_value>,

"login_name":<String_value>,

"id":<String_value>,

"ip_address":<String_value>,

"nested_group_extraction":<Boolean_value>,

"group_search_attribute":<String_value>,

"ssh_public_key":<String_value>,

"group_name_identifier":<String_value>,

"subattribute_name":<String_value>,

"name":<String_value>,

"validate_ldap_server_certs":<Boolean_value>,

"group_search_filter":<String_value>,

"bind_dn":<String_value>,

"port":<Integer_value>,
```

```
"base_dn":<String_value>,

"sec_type":<String_value>,

"max_ldap_referrals":<Integer_value>,

"type":<String_value>,

"bind_passwd":<String_value>,

"address_type":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ldap_server/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ldap_server":[{

"auth_timeout":<Integer_value>,

"group_search_subattribute":<String_value>,
```

```
"follow_referrals":<Boolean_value>,

"search_filter":<String_value>,

"group_attr_name":<String_value>,

"authentication":<Boolean_value>,

"ldap_host_name":<String_value>,

"change_password":<Boolean_value>,

"default_authentication_group":<String_value>,

"max_nesting_level":<Integer_value>,

"login_name":<String_value>,

"id":<String_value>,

"ip_address":<String_value>,

"nested_group_extraction":<Boolean_value>,

"group_search_attribute":<String_value>,

"ssh_public_key":<String_value>,

"group_name_identifier":<String_value>,

"subattribute_name":<String_value>,

"name":<String_value>,

"validate_ldap_server_certs":<Boolean_value>,

"group_search_filter":<String_value>,

"bind_dn":<String_value>,

"port":<Integer_value>,

"base_dn":<String_value>,
```

```
"sec_type":<String_value>,

"max_ldap_referrals":<Integer_value>,

"type":<String_value>,

"bind_passwd":<String_value>,

"address_type":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/ldap_server/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ldap_server":{

<b>"id":<String_value></b>,

<b>"ip_address":<String_value></b>,

<b>"name":<String_value></b>,

"auth_timeout":<Integer_value>,

"search_filter":<String_value>,

"ldap_host_name":<String_value>,

"authentication":<Boolean_value>,
```

```
"change_password":<Boolean_value>,

"default_authentication_group":<String_value>,

"login_name":<String_value>,

"group_name_identifier":<String_value>,

"bind_dn":<String_value>,

"port":<Integer_value>,

"sec_type":<String_value>,

"base_dn":<String_value>,

"bind_passwd":<String_value>,

"type":<String_value>,

"group_search_subattribute":<String_value>,

"follow_referrals":<Boolean_value>,

"group_attr_name":<String_value>,

"max_nesting_level":<Integer_value>,

"nested_group_extraction":<Boolean_value>,

"group_search_attribute":<String_value>,

"ssh_public_key":<String_value>,

"subattribute_name":<String_value>,

"validate_ldap_server_certs":<Boolean_value>,

"group_search_filter":<String_value>,

"max_ldap_referrals":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ldap_server":[{

"auth_timeout":<Integer_value>,

"group_search_subattribute":<String_value>,

"follow_referrals":<Boolean_value>,

"search_filter":<String_value>,

"group_attr_name":<String_value>,

"authentication":<Boolean_value>,

"ldap_host_name":<String_value>,

"change_password":<Boolean_value>,

"default_authentication_group":<String_value>,

"max_nesting_level":<Integer_value>,

"login_name":<String_value>,

"id":<String_value>,

"ip_address":<String_value>,

"nested_group_extraction":<Boolean_value>,

"group_search_attribute":<String_value>,

"ssh_public_key":<String_value>,
```

```
"group_name_identifier":<String_value>,

"subattribute_name":<String_value>,

"name":<String_value>,

"validate_ldap_server_certs":<Boolean_value>,

"group_search_filter":<String_value>,

"bind_dn":<String_value>,

"port":<Integer_value>,

"base_dn":<String_value>,

"sec_type":<String_value>,

"max_ldap_referrals":<Integer_value>,

"type":<String_value>,

"bind_passwd":<String_value>,

"address_type":<Integer_value>}]}
```

login

Configuration for Login/Logout resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ID	<String>	Read-write	ID of the User that wants to login in cloud. Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
session_timeout	<Integer>	Read-write	Session Timeout in seconds, if no activity is performed for specified time then session will be expired.
auth_with_remote_pubkey	<Integer>	Read-write	Login request to specify if SSH authentication took place with remote public keys for this user, 1 for remotely managed pubkey else 0.
password	<String>	Read-write	Password. Maximum length = 128
challenge	<String>	Read-write	Challenge request for authentication.
challenge_response	<String>	Read-write	Challenge response for authentication.
client_port	<Integer>	Read-write	Login Client port. Maximum value =
token	<String>	Read-write	Random token to identify session during logout.
permission	<String>	Read-write	Actions that this user is authorized to perform. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
username	<String>	Read-write	User Name that wants to login. Maximum length = 128
client_ip	<String>	Read-write	Login client IP.
session_creator	<String>	Read-write	session_creator.
sessionid	<String>	Read-write	Session ID would only be set if login was performed successfully.
Secret	<String>	Read-write	Secret. Maximum length = 128

Operations

(click to see Properties)

- ADD
- DELETE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/login?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"login": {

<b>"password":<String_value></b>,

<b>"username":<String_value></b>,

"ID":<String_value>,

"session_timeout":<Integer_value>,
```

```
"auth_with_remote_pubkey":<Integer_value>,

"challenge_response":<String_value>,

"challenge":<String_value>,

"client_port":<Integer_value>,

"token":<String_value>,

"permission":<String_value>,

"session_creator":<String_value>,

"client_ip":<String_value>,

"sessionid":<String_value>,

"Secret":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"login":[{

"ID":<String_value>,

"session_timeout":<Integer_value>,

"auth_with_remote_pubkey":<Integer_value>,

"login_type":<String_value>,

"password":<String_value>,

"challenge":<String_value>,
```

```
"challenge_response":<String_value>,

"client_port":<Integer_value>,

"token":<String_value>,

"tenant_name":<String_value>,

"permission":<String_value>,

"challenge_token":<String_value>,

"username":<String_value>,

"client_ip":<String_value>,

"session_creator":<String_value>,

"sessionid":<String_value>,

"cert_verified":<Boolean_value>,

"Secret":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/login/ **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

logout

Configuration for Logout resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
sessionid	<String>	Read-write	Session ID would only be set if login was performed successfully.

Operations

(click to see Properties)

- DELETE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/logout?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"logout": {

<b>"sessionid":<String_value></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"logout":[{

"password":<String_value>,

"session_timeout":<Integer_value>,

"sessionid":<String_value>,

"username":<String_value>}]}
```

mail_attachment

Configuration for Attachments for e-mail action configured in filter rules resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Double>	Read-only	Size of the file attached.
file_last_modified	<String>	Read-only	File Last Modified.

Operations

(click to see Properties)

- UPLOAD

- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/mail_attachment/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/mail_attachment **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/mail_attachment **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of mail_attachment resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/mail_attachment **? pagesize=#no;pageno=#no** Use this query-parameter to get the mail_attachment resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/mail_attachment **?count=yes** Use this query-parameter to get the count of mail_attachment resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mail_attachment":[{

"file_location_path":<String_value>,

"file_size":<Double_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_dir":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/mail_attachment/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mail_attachment":[{

"file_location_path":<String_value>,

"file_size":<Double_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_dir":<Boolean_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/mail_attachment **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

mail_profile

Configuration for Mail profile resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
to_list	<String>	Read-write	List of to whom send the mail. . Minimum length = 6
sender_mail_address	<String>	Read-write	Email Address from where mail is send.
bcc_list	<String>	Read-write	List to whom BCC the mail..
id	<String>	Read-write	Id is system generated key for all the mail profile..
cc_list	<String>	Read-write	List to whom CC the mail..
server_name	<String>	Read-write	SMTP server name. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
profile_name	<String>	Read-write	Profile name for the mail setting.. Minimum length = 1 Maximum length = 128

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/mail_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"mail_profile": {

<b>"to_list":<String_value></b>,

<b>"server_name":<String_value></b>,

<b>"profile_name":<String_value></b>,

"bcc_list":<String_value>,

"id":<String_value>,

"cc_list":<String_value>,

"sender_mail_address":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mail_profile":[{

"tenant_id":<String_value>,

"to_list":<String_value>,

"sender_mail_address":<String_value>,

"bcc_list":<String_value>,

"id":<String_value>,

"cc_list":<String_value>,

"server_name":<String_value>,

"profile_name":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/mail_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/mail_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/mail_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of mail_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/mail_profile **? pagesize=#no;pageno=#no** Use this query-parameter to get the mail_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/mail_profile **?count=yes** Use this query-parameter to get the count of mail_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mail_profile":[{

"tenant_id":<String_value>,

"to_list":<String_value>,

"sender_mail_address":<String_value>,

"bcc_list":<String_value>,

"id":<String_value>,

"cc_list":<String_value>,

"server_name":<String_value>,

"profile_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/mail_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mail_profile":[{

"tenant_id":<String_value>,

"to_list":<String_value>,

"sender_mail_address":<String_value>,

"bcc_list":<String_value>,

"id":<String_value>,

"cc_list":<String_value>,

"server_name":<String_value>,

"profile_name":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/mail_profile/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"mail_profile":{

<b>"id":<String_value></b>,

"bcc_list":<String_value>,

"to_list":<String_value>,
```

```
"server_name":<String_value>,

"cc_list":<String_value>,

"sender_mail_address":<String_value>,

"profile_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mail_profile":[{

"tenant_id":<String_value>,

"to_list":<String_value>,

"sender_mail_address":<String_value>,

"bcc_list":<String_value>,

"id":<String_value>,

"cc_list":<String_value>,

"server_name":<String_value>,

"profile_name":<String_value>}]}
```

managed_vserver

Configuration for Managed Vserver Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
managed	<Boolean>	Read-write	Managed.
type	<String>	Read-write	Vserver Type. Minimum length = 1
id	<String>	Read-write	Id of the Vserver.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

metering_data

Configuration for Metering Data for VM Devices resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the metering data points.
ip_address	<String>	Read-write	IP Address for this VM device for which data is recorded. Minimum length = 1 Maximum length = 64
interval_txmbits	<Double>	Read-only	Out Tx of VM Instance for Interval in Mbits.
burst_thpt	<Double>	Read-only	Burst Throughput of VM Instance in Mbps.
tx	<Double>	Read-only	Out Throughput of VM Instance in Mbps.
rx	<Double>	Read-only	In Throughput of VM Instance in Mbps.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
start_time	<Double>	Read-only	Start Time of Interval for which data was recorded.
min_thpt	<Double>	Read-only	Minimum Throughput allocated to Instance in Mbps.
interval_rxmbits	<Double>	Read-only	In Rx of VM Instance for Interval in Mbits.
poll_time	<Double>	Read-only	Poll Time at which data was recorded.
max_thpt_limit	<Double>	Read-only	Maximum Throughput Limit allowed for VM Instance in Mbps.
total_txmbits	<Double>	Read-only	Out Total Rx of VM Instance in Mbits.
thpt_limit	<Double>	Read-only	Throughput Limit configured for VM Instance in Mbps.
total_rxmbits	<Double>	Read-only	In Total Rx of VM Instance in Mbits.

Operations

(click to see Properties)

- PRUNE
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

prune

URL: https://<MGMT-IP>/nitro/v2/config/metering_data/id_value<String>?action=prune;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"metering_data": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/metering_data **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/metering_data **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of metering_data resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/metering_data **? pagesize=#no;pageno=#no** Use this query-parameter to get the metering_data resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/metering_data **?count=yes** Use this query-parameter to get the count of metering_data resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"metering_data":[{

"interval_txmbits":<Double_value>,

"burst_thpt":<Double_value>,

"tx":<Double_value>,

"rx":<Double_value>,

"start_time":<Double_value>,

"min_thpt":<Double_value>,

"interval_rxmbits":<Double_value>,

"poll_time":<Double_value>,

"max_thpt_limit":<Double_value>,

"total_txmbits":<Double_value>,

"thpt_limit":<Double_value>,

"total_rxmbits":<Double_value>,

"id":<String_value>,

"ip_address":<String_value>}]}
```

metering_export

Configuration for Metering Data Export resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 512
ip_address	<String>	Read-write	IP Address for this VM device for which metering data export is required. Minimum length = 1 Maximum length = 64
end_time	<Double>	Read-only	End Time of interval for which export is generated.
file_size	<Integer>	Read-only	file_size.
start_time	<Double>	Read-only	Start Time of interval for which export is generated.
file_last_modified	<String>	Read-only	Last Modified.
duration	<String>	Read-only	Interval Duration for which export is generated.

Operations

(click to see Properties)

- DELETE
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/metering_export/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/metering_export **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/metering_export **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of metering_export resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/metering_export **? pagesize=#no;pageno=#no** Use this query-parameter to get the metering_export resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/metering_export **?count=yes** Use this query-parameter to get the count of metering_export resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"metering_export":[{

"end_time":<Double_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"start_time":<Double_value>,

"file_last_modified":<String_value>,

"duration":<String_value>,

"ip_address":<String_value>}]}
```

metering_summary

Configuration for Metering Data Summary resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
number_of_bursts	<Integer>	Read-write	Number of bursts by VM Instance in Interval.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address	<String>	Read-write	IP Address for this VM device for which metering data summary is required. Minimum length = 1 Maximum length = 64
max_burst	<Double>	Read-only	Maximum Burst Throughput of VM Instance for Interval in Mbps.
min_rx	<Double>	Read-only	Minimum In Throughput of VM Instance for Interval in Mbps.
start_time	<Double>	Read-only	Start Time of interval for which summary is generated.
duration	<String>	Read-only	Interval Duration for which summary is generated.
total_txmbits	<Double>	Read-only	Out Total Rx of VM Instance for Interval in Mbits.
total_rxmbits	<Double>	Read-only	In Total Rx of VM Instance for Interval in Mbits.
end_time	<Double>	Read-only	End Time of interval for which summary is generated.
max_rx	<Double>	Read-only	Maximum In Throughput of VM

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Instance for Interval in Mbps.
max_tx	<Double>	Read-only	Maximum Out Throughput of VM Instance for Interval in Mbps.
min_tx	<Double>	Read-only	Minimum Out Throughput of VM Instance for Interval in Mbps.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/metering_summary **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/metering_summary **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of metering_summary resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/metering_summary **? pagesize=#no;pageno=#no** Use this query-parameter to get the metering_summary resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/metering_summary **?count=yes** Use this query-parameter to get the count of metering_summary resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"metering_summary":[{

"max_burst":<Double_value>,

"min_rx":<Double_value>,

"start_time":<Double_value>,

"duration":<String_value>,

"number_of_bursts":<Integer_value>,

"total_txmbits":<Double_value>,

"total_rxmbits":<Double_value>,

"end_time":<Double_value>,

"max_rx":<Double_value>,

"max_tx":<Double_value>,

"min_tx":<Double_value>,

"ip_address":<String_value>}]}
```

mps

Configuration for System Status resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
maps_apikey	<String>	Read-write	Maps API Key.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
deployment_type	<String>	Read-write	Indicates the type of deployment of Citrix ADM: standalone or ha or scaleout..
is_passive	<Boolean>	Read-write	Indicates the node's state: ACTIVE or PASSIVE in a HA deployment..
privilege_feature	<String>	Read-write	privilege_feature.
is_cloud	<Boolean>	Read-write	True if its a cloud deployment.
is_member_of_default_group	<Boolean>	Read-write	Is Member Of Default Group.
motd	<String>	Read-write	Message of the Day (contents of motd file) that can be shown on UI after successful login.
config_motd	<Boolean>	Read-write	Configure Message of the Day (contents of motd file), this needs to be set true if user wants to configure message if the day.
is_container	<Boolean>	Read-write	True if its a container deployment.
hist_mig_inprog	<Boolean>	Read-write	Indicates whether the historical tables database migration is in progress or not..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
upgrade_agent_version	<String>	Read-write	Indicates the next version the agent needs to upgrade to..
build_number_short	<String>	Read-only	Build Number without Date.
sysid	<String>	Read-only	System Id.
hostname	<String>	Read-only	Host name on which system is running.
bios_version	<String>	Read-only	BIOS Version.
uptime	<String>	Read-only	Uptime.
current_user_permission	<String>	Read-only	This property will show the permission type for current user.
product_build_number	<String>	Read-only	Product Build Number.
serial	<String>	Read-only	Serial Number.
username	<String>	Read-only	User Name who is currently connected to the system.
current_tenant	<String>	Read-only	Current Tenant Name for logged-in user.
current_time_formatted	<String>	Read-only	Current Time (Formatted).
current_time	<Integer>	Read-only	Current Time.
hostid	<String>	Read-only	Host Id.
product_version	<String>	Read-only	Product Version.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
platform	<String>	Read-only	Platform.
hypervisor_uptime	<String>	Read-only	Hypervisor Uptime.
host	<String>	Read-only	Host IP Address on which system is running, this will set for each client session only.
build_number	<String>	Read-only	Build Number.
time_zone_offset	<Integer>	Read-only	Time zone offset in minutes (Example:-330).
time_zone	<String>	Read-only	Server Time Zone.
product	<String>	Read-only	Product Name.
current_tenant_id	<String>	Read-only	Current Tenant Id for logged-in user.

Operations

(click to see Properties)

- GET
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get

URL: https://<MGMT-IP>/nitro/v2/config/mps **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps":[{

"maps_apikey":<String_value>,

"build_number_short":<String_value>,

"sysid":<String_value>,

"hostname":<String_value>,

"bios_version":<String_value>,

"deployment_type":<String_value>,

"uptime":<String_value>,

"current_user_permission":<String_value>,

"is_passive":<Boolean_value>,

"product_build_number":<String_value>,

"serial":<String_value>,

"privilege_feature":<String_value>,

"username":<String_value>,

"current_tenant":<String_value>,

"current_time_formatted":<String_value>,

"is_cloud":<Boolean_value>,

"current_time":<Integer_value>,

"hostid":<String_value>,
```

```
"is_member_of_default_group":<Boolean_value>,

"product_version":<String_value>,

"platform":<String_value>,

"motd":<String_value>,

"hypervisor_uptime":<String_value>,

"config_motd":<Boolean_value>,

"host":<String_value>,

"build_number":<String_value>,

"time_zone_offset":<Integer_value>,

"is_container":<Boolean_value>,

"hist_mig_inprog":<Boolean_value>,

"time_zone":<String_value>,

"product":<String_value>,

"upgrade_agent_version":<String_value>,

"current_tenant_id":<String_value>}]}
```

modify

URL: https://<MGMT-IP>/nitro/v2/config/mps/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"mps":{

"maps_apikey":<String_value>,

"is_cloud":<Boolean_value>,

"is_passive":<Boolean_value>,

"config_motd":<Boolean_value>,

"privilege_feature":<String_value>,

"is_member_of_default_group":<Boolean_value>,

"deployment_type":<String_value>,

"is_container":<Boolean_value>,

"hist_mig_inprog":<Boolean_value>,

"upgrade_agent_version":<String_value>,

"motd":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps":[{

"maps_apikey":<String_value>,

"build_number_short":<String_value>,

"sysid":<String_value>,
```

```
"hostname":<String_value>,

"bios_version":<String_value>,

"deployment_type":<String_value>,

"uptime":<String_value>,

"current_user_permission":<String_value>,

"is_passive":<Boolean_value>,

"product_build_number":<String_value>,

"serial":<String_value>,

"privilege_feature":<String_value>,

"username":<String_value>,

"current_tenant":<String_value>,

"current_time_formatted":<String_value>,

"is_cloud":<Boolean_value>,

"current_time":<Integer_value>,

"hostid":<String_value>,

"is_member_of_default_group":<Boolean_value>,

"product_version":<String_value>,

"platform":<String_value>,

"motd":<String_value>,

"hypervisor_uptime":<String_value>,

"config_motd":<Boolean_value>,

"host":<String_value>,
```

```
"build_number":<String_value>,

"time_zone_offset":<Integer_value>,

"is_container":<Boolean_value>,

"hist_mig_inprog":<Boolean_value>,

"time_zone":<String_value>,

"product":<String_value>,

"upgrade_agent_version":<String_value>,

"current_tenant_id":<String_value>}]}
```

mps_disk_usage

Configuration for Available Disks Usage resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_name	<String>	Read-write	.
parent_id	<String>	Read-write	.
disk_partition	<String>	Read-write	Disk Partition.
id	<String>	Read-write	.
free_space	<Double>	Read-only	Free space available in KB.
disk_usage	<Double>	Read-only	Disk Usage (%).
used_space	<Double>	Read-only	Used space available in KB.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
disk_name	<String>	Read-only	Disk Name.
total_space	<Double>	Read-only	Total space in KB.
disk_slice	<String>	Read-only	Disk Slice.
disk_capacity	<Double>	Read-only	Disk Total Capacity in KB.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/mps_disk_usage **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/mps_disk_usage **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of mps_disk_usage resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/mps_disk_usage **? pagesize=#no;pageno=#no** Use this query-parameter to get the mps_disk_usage resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/mps_disk_usage **?count=yes** Use this query-parameter to get the count of mps_disk_usage resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_disk_usage":[{

"free_space":<Double_value>,

"parent_name":<String_value>,

"disk_usage":<Double_value>,

"used_space":<Double_value>,

"parent_id":<String_value>,

"disk_name":<String_value>,

"total_space":<Double_value>,

"disk_partition":<String_value>,

"disk_slice":<String_value>,

"id":<String_value>,

"disk_capacity":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/mps_disk_usage/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_disk_usage":[{

"free_space":<Double_value>,

"parent_name":<String_value>,

"disk_usage":<Double_value>,

"used_space":<Double_value>,

"parent_id":<String_value>,

"disk_name":<String_value>,

"total_space":<Double_value>,

"disk_partition":<String_value>,

"disk_slice":<String_value>,

"id":<String_value>,

"disk_capacity":<Double_value>}]}
```

mps_feature

Configuration for Feature Toggle Status resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
admin_toggle	<Integer>	Read-write	This is Admin controllable. 0: Disable UI and Backend, 1: Disable UI and enable Backend, 2: Enable UI and disable Backend, 3: Enable UI and Backend.. Maximum value =
ops_toggle	<Integer>	Read-write	This is Ops controllable. 0: Disable UI and Backend, 1: Disable UI and enable Backend, 2: Enable UI and disable Backend, 3: Enable UI and Backend. Ops controlled takes higher precedence than Admin Controlled.. Maximum value =
built_in	<Boolean>	Read-write	This is Ops controllable and will not be visible to the Admin to control. If true: Ops controllable feature, false: Admin controllable feature..
feature_name	<String>	Read-write	Feature Name.
id	<String>	Read-write	Feature ID.
description	<String>	Read-write	Feature Description..

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/mps_feature?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"mps_feature": {

"id":<String_value>,

"description":<String_value>,

"admin_toggle":<Integer_value>,

"ops_toggle":<Integer_value>,

"built_in":<Boolean_value>,

"feature_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_feature":[{

"admin_toggle":<Integer_value>,
```

```
"ops_toggle":<Integer_value>,

"built_in":<Boolean_value>,

"feature_name":<String_value>,

"id":<String_value>,

"description":<String_value>,

"info_from_cloud":<Boolean_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/mps_feature/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/mps_feature **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/mps_feature **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of mps_feature resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/mps_feature **?pagesize=#no;pageno=#no** Use this query-parameter to get the mps_feature resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/mps_feature **?count=yes** Use this query-parameter to get the count of mps_feature resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_feature":[{

"admin_toggle":<Integer_value>,

"ops_toggle":<Integer_value>,

"built_in":<Boolean_value>,

"feature_name":<String_value>,

"id":<String_value>,

"description":<String_value>,
```

```
"info_from_cloud":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/mps_feature/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_feature":[{

"admin_toggle":<Integer_value>,

"ops_toggle":<Integer_value>,

"built_in":<Boolean_value>,

"feature_name":<String_value>,

"id":<String_value>,

"description":<String_value>,

"info_from_cloud":<Boolean_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/mps_feature/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"mps_feature":{

<b>"id":<String_value></b>,

"description":<String_value>,

"admin_toggle":<Integer_value>,

"ops_toggle":<Integer_value>,

"built_in":<Boolean_value>,

"feature_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_feature":[{

"admin_toggle":<Integer_value>,
```

```
"ops_toggle":<Integer_value>,

"built_in":<Boolean_value>,

"feature_name":<String_value>,

"id":<String_value>,

"description":<String_value>,

"info_from_cloud":<Boolean_value>}]}
```

mps_health

Configuration for Health Monitoring Stats resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
disks_usage	<mps_disk_usage[]>	Read-write	Individual Disks Usage.
node_type	<String>	Read-write	Application Node Type.
node_id	<String>	Read-write	Application Node ID.
id	<String>	Read-write	Id is system generated key.
memory_total	<Double>	Read-only	Total memory in KB.
disk_total_capacity	<Double>	Read-only	Disk Total Capacity in KB.
disk_usage	<Double>	Read-only	Disk Usage (%).
disk_total	<Double>	Read-only	Total available disk space in KB.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
memory_usage	<Double>	Read-only	Memory Usage (%).
disk_free	<Double>	Read-only	Free disk available in KB.
page_size	<Double>	Read-only	Total memory in KB.
memory_free	<Double>	Read-only	Free memory available in KB.
cpu_usage	<Double>	Read-only	CPU Usage (%).
disk_used	<Double>	Read-only	Disk used in KB.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/mps_health **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/mps_health **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of mps_health resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/mps_health **? pagesize=#no;pageno=#no** Use this query-parameter to get the mps_health resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/mps_health **?count=yes** Use this query-parameter to get the count of mps_health resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_health":[{

"disks_usage":[{

"free_space":<Double_value>,

"parent_name":<String_value>,

"disk_usage":<Double_value>,

"used_space":<Double_value>,

"parent_id":<String_value>,

"disk_name":<String_value>,

"total_space":<Double_value>,

"disk_partition":<String_value>,

"disk_slice":<String_value>,

"id":<String_value>,

"disk_capacity":<Double_value>}],

"node_type":<String_value>,

"memory_total":<Double_value>,

"disk_total_capacity":<Double_value>,

"disk_usage":<Double_value>,

"disk_total":<Double_value>,
```

```
"memory_usage":<Double_value>,

"disk_free":<Double_value>,

"page_size":<Double_value>,

"node_id":<String_value>,

"memory_free":<Double_value>,

"cpu_usage":<Double_value>,

"id":<String_value>,

"disk_used":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/mps_health/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_health":[{

"disks_usage":[{

"free_space":<Double_value>,

"parent_name":<String_value>,

"disk_usage":<Double_value>,

"used_space":<Double_value>,

"parent_id":<String_value>,

"disk_name":<String_value>,

"total_space":<Double_value>,

"disk_partition":<String_value>,

"disk_slice":<String_value>,

"id":<String_value>,

"disk_capacity":<Double_value>}],

"node_type":<String_value>,

"memory_total":<Double_value>,

"disk_total_capacity":<Double_value>,

"disk_usage":<Double_value>,

"disk_total":<Double_value>,

"memory_usage":<Double_value>,

"disk_free":<Double_value>,

"page_size":<Double_value>,
```

```
"node_id":<String_value>,

"memory_free":<Double_value>,

"cpu_usage":<Double_value>,

"id":<String_value>,

"disk_used":<Double_value>}]}
```

mps_image

Configuration for System Build File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file size in MB.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/mps_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/mps_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/mps_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of mps_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/mps_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the mps_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/mps_image **?count=yes** Use this query-parameter to get the count of mps_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_image":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_dir":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/mps_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_image":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_dir":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/mps_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

mps_ssl_certkey

Configuration for Install SSL certificate on Management Service resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
password	<String>	Read-write	The pass-phrase that was used to encrypt the private-key.. Maximum length = 32
ssl_key	<String>	Read-write	Key. Minimum length = 1 Maximum length = 128
ssl_certificate	<String>	Read-write	Certificate. Minimum length = 1 Maximum length = 128
fingerprint	<String>	Read-write	SHA-1 Fingerprint of Citrix ADM SSL Certificate. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 512
serial_number	<String>	Read-only	Serial Number.
signature_algorithm	<String>	Read-only	Signature Algorithm.
valid_from	<String>	Read-only	Valid From.
status	<String>	Read-only	Tells whether the certificate is still valid or not.
issuer	<String>	Read-only	Issuer.
public_key_size	<Integer>	Read-only	Public Key Size.
valid_to	<String>	Read-only	Valid To.
version	<String>	Read-only	Version.
subject	<String>	Read-only	Subject.
public_key_algorithm	<String>	Read-only	Public Key Algorithm.
days_to_expiry	<Integer>	Read-only	Days before SSL certificate expires.

Operations

(click to see Properties)

- ADD
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/mps_ssl_certkey?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"mps_ssl_certkey": {

<b>"ssl_certificate":<String_value></b>,

"fingerprint":<String_value>,

"password":<String_value>,

"ssl_key":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_ssl_certkey":[{

"serial_number":<String_value>,

"signature_algorithm":<String_value>,

"valid_from":<String_value>,

"status":<String_value>,

"issuer":<String_value>,

"public_key_size":<Integer_value>,

"password":<String_value>,

"ssl_key":<String_value>,

"valid_to":<String_value>,

"reboot":<Boolean_value>,

"version":<String_value>,
```

```
"subject":<String_value>,

"public_key_algorithm":<String_value>,

"days_to_expiry":<Integer_value>,

"ssl_certificate":<String_value>,

"cert_format":<String_value>,

"fingerprint":<String_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/mps_ssl_certkey **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/mps_ssl_certkey **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of mps_ssl_certkey resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/mps_ssl_certkey **? pagesize=#no;pageno=#no** Use this query-parameter to get the mps_ssl_certkey resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/mps_ssl_certkey **?count=yes** Use this query-parameter to get the count of mps_ssl_certkey resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_ssl_certkey":[{

"serial_number":<String_value>,

"signature_algorithm":<String_value>,

"valid_from":<String_value>,

"status":<String_value>,

"issuer":<String_value>,

"public_key_size":<Integer_value>,

"password":<String_value>,

"ssl_key":<String_value>,

"valid_to":<String_value>,

"reboot":<Boolean_value>,

"version":<String_value>,

"subject":<String_value>,

"public_key_algorithm":<String_value>,

"days_to_expiry":<Integer_value>,

"ssl_certificate":<String_value>,
```

```
"cert_format":<String_value>,

"fingerprint":<String_value>}]}
```

mps_upgrade

Configuration for System Upgrade resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	image_name. Minimum length = 1 Maximum length = 128
doc_file	<String>	Read-write	This property is no longer used;ltbr;gtDocumentation File Name. Minimum length = 1 Maximum length = 128
do_cleanup	<Boolean>	Read-write	True, if user wants to clean the image files once upgrade is completed successfully.

Operations

(click to see Properties)

- UPGRADE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

upgrade

URL: https://<MGMT-IP>/nitro/v2/config/mps_upgrade?action=upgrade;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"mps_upgrade": {

<b>"image_name":<String_value></b>,

"do_cleanup":<Boolean_value>,

"doc_file":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mps_upgrade":[{

"image_name":<String_value>,

"doc_file":<String_value>,

"do_cleanup":<Boolean_value>}]}
```

mpsgroup

Configuration for System Groups resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
session_timeout	<Integer>	Read-write	Session timeout for the Group.
session_timeout_unit	<String>	Read-write	Session timeout unit for the Group.
assign_all_apps	<Boolean>	Read-write	Assign All Applications (YES|NO).
enable_session_timeout	<Boolean>	Read-write	Enables session timeout for group.
select_individual_entity	<Boolean>	Read-write	Select Individual Entity Type.
tenant_id	<String>	Read-write	Id of the tenant. Minimum length = 1 Maximum length = 128
apply_all_bound_entities	<Boolean>	Read-write	Apply for all bound entities (TRUE|FALSE).
assign_all_devices	<Boolean>	Read-write	Assign All Instances (YES|NO).
assign_all_selected_device_apps	<Boolean>	Read-write	Assign All Application from selected instances (YES|NO).
id	<String>	Read-write	Id is system generated key for all the system groups.
allow_application_only	<Boolean>	Read-write	Checks if only application centic page is needed.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Group Name. Minimum length = 1 Maximum length = 64
permission	<String>	Read-write	Permission for the group (admin/read-only). Minimum length = 1 Maximum length = 128
description	<String>	Read-write	Description of Group. Minimum length = 1 Maximum length = 1024
bound_entity_selected	<Integer>	Read-write	Which bound entiy is selected VSERVER(0),SERVICE(1),SERVICEGROUP(2),SERVER(3).
assign_all_autoscale_groups	<Boolean>	Read-write	Assign All Autoscale groups (YES|NO).
role	<String>	Read-write	Role (admin|nonadmin).
roles	<String[]>	Read-write	Roles assigned to the group.
standalone_instances_id	<String[]>	Read-write	Stand alone instances belong to this groupp.
users	<String[]>	Read-write	Users belong to the group.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
application_names	<String[]>	Read-write	All Application names that are part of this group.This includes selected appnames as well as applications which are result of defined regex.
autoscale_groups_id	<String[]>	Read-write	Autoscale groups belong to this groupp.
application_names_without_regex	<String[]>	Read-only	selected application names that are part of this group.
application_names_with_regex	<String[]>	Read-only	Application names defined with regex that are part of this group.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/mpsgroup?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"mpsgroup": {

<b>"name":<String_value></b>,

<b>"permission":<String_value></b>,

"session_timeout":<Integer_value>,

"session_timeout_unit":<String_value>,

"assign_all_apps":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"id":<String_value>,

"autoscale_groups_id":<String_value[]>,

"roles":<String_value[]>,

"allow_application_only":<Boolean_value>,

"description":<String_value>,

"application_names":<String_value[]>,

"assign_all_autoscale_groups":<Boolean_value>,

"role":<String_value>,

"standalone_instances_id":<String_value[]>,

"users":<String_value[]>,

"tenant_id":<String_value>,

"select_individual_entity":<Boolean_value>,

"assign_all_devices":<Boolean_value>,

"apply_all_bound_entities":<Boolean_value>,

"assign_all_selected_device_apps":<Boolean_value>,
```

```
"bound_entity_selected":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mpsgroup":[{

"session_timeout":<Integer_value>,

"session_timeout_unit":<String_value>,

"assign_all_apps":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"select_individual_entity":<Boolean_value>,

"tenant_id":<String_value>,

"apply_all_bound_entities":<Boolean_value>,

"assign_all_devices":<Boolean_value>,

"assign_all_selected_device_apps":<Boolean_value>,

"id":<String_value>,

"allow_application_only":<Boolean_value>,

"name":<String_value>,

"permission":<String_value>,

"description":<String_value>,

"bound_entity_selected":<Integer_value>,
```

```
"assign_all_autoscale_groups":<Boolean_value>,

"role":<String_value>,

"roles":<String_value>,

"application_names_without_regex":<String_value>,

"tenant_name":<String_value>,

"standalone_instances_id":<String_value>,

"users":<String_value>,

"application_names":<String_value>,

"application_list":[{

"name":<String_value>,

"display_name":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"vsvr_ip_address":<String_value>,

"configuration_path":<String_value>,

"id":<String_value>,

"type":<String_value>}],

"application_names_with_regex":<String_value>,

"autoscale_groups_id":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/mpsgroup/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/mpsgroup **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/mpsgroup **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of mpsgroup resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/mpsgroup **? pagesize=#no;pageno=#no** Use this query-parameter to get the mpsgroup resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/mpsgroup **?count=yes** Use this query-parameter to get the count of mpsgroup resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mpsgroup":[{

"session_timeout":<Integer_value>,

"session_timeout_unit":<String_value>,

"assign_all_apps":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"select_individual_entity":<Boolean_value>,

"tenant_id":<String_value>,

"apply_all_bound_entities":<Boolean_value>,

"assign_all_devices":<Boolean_value>,

"assign_all_selected_device_apps":<Boolean_value>,

"id":<String_value>,

"allow_application_only":<Boolean_value>,

"name":<String_value>,

"permission":<String_value>,

"description":<String_value>,

"bound_entity_selected":<Integer_value>,
```

```
"assign_all_autoscale_groups":<Boolean_value>,

"role":<String_value>,

"roles":<String_value>,

"application_names_without_regex":<String_value>,

"tenant_name":<String_value>,

"standalone_instances_id":<String_value>,

"users":<String_value>,

"application_names":<String_value>,

"application_list":[{

"name":<String_value>,

"display_name":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"vsvr_ip_address":<String_value>,

"configuration_path":<String_value>,

"id":<String_value>,

"type":<String_value>}],

"application_names_with_regex":<String_value>,

"autoscale_groups_id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/mpsgroup/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mpsgroup":[{

"session_timeout":<Integer_value>,

"session_timeout_unit":<String_value>,

"assign_all_apps":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"select_individual_entity":<Boolean_value>,

"tenant_id":<String_value>,

"apply_all_bound_entities":<Boolean_value>,

"assign_all_devices":<Boolean_value>,

"assign_all_selected_device_apps":<Boolean_value>,

"id":<String_value>,

"allow_application_only":<Boolean_value>,
```

```
"name":<String_value>,

"permission":<String_value>,

"description":<String_value>,

"bound_entity_selected":<Integer_value>,

"assign_all_autoscale_groups":<Boolean_value>,

"role":<String_value>,

"roles":<String_value>,

"application_names_without_regex":<String_value>,

"tenant_name":<String_value>,

"standalone_instances_id":<String_value>,

"users":<String_value>,

"application_names":<String_value>,

"application_list":[{

"name":<String_value>,

"display_name":<String_value>,

"state":<String_value>,

"ns_ip_address":<String_value>,

"vsvr_ip_address":<String_value>,

"configuration_path":<String_value>,

"id":<String_value>,

"type":<String_value>}],

"application_names_with_regex":<String_value>,
```

```
"autoscale_groups_id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/mpsgroup/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"mpsgroup":{

<b>"id":<String_value></b>,

"session_timeout":<Integer_value>,

"session_timeout_unit":<String_value>,

"assign_all_apps":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"autoscale_groups_id":<String_value[]>,

"roles":<String_value[]>,

"allow_application_only":<Boolean_value>,

"permission":<String_value>,

"name":<String_value>,

"description":<String_value>,
```

```
"application_names":<String_value[]>,

"assign_all_autoscale_groups":<Boolean_value>,

"role":<String_value>,

"standalone_instances_id":<String_value[]>,

"users":<String_value[]>,

"tenant_id":<String_value>,

"select_individual_entity":<Boolean_value>,

"assign_all_devices":<Boolean_value>,

"apply_all_bound_entities":<Boolean_value>,

"assign_all_selected_device_apps":<Boolean_value>,

"bound_entity_selected":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mpsgroup":[{

"session_timeout":<Integer_value>,

"session_timeout_unit":<String_value>,

"assign_all_apps":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"select_individual_entity":<Boolean_value>,
```

```
"tenant_id":<String_value>,

"apply_all_bound_entities":<Boolean_value>,

"assign_all_devices":<Boolean_value>,

"assign_all_selected_device_apps":<Boolean_value>,

"id":<String_value>,

"allow_application_only":<Boolean_value>,

"name":<String_value>,

"permission":<String_value>,

"description":<String_value>,

"bound_entity_selected":<Integer_value>,

"assign_all_autoscale_groups":<Boolean_value>,

"role":<String_value>,

"roles":<String_value>,

"application_names_without_regex":<String_value>,

"tenant_name":<String_value>,

"standalone_instances_id":<String_value>,

"users":<String_value>,

"application_names":<String_value>,

"application_list":[{

"name":<String_value>,

"display_name":<String_value>,

"state":<String_value>,
```

```
"ns_ip_address":<String_value>,

"vsvr_ip_address":<String_value>,

"configuration_path":<String_value>,

"id":<String_value>,

"type":<String_value>}],

"application_names_with_regex":<String_value>,

"autoscale_groups_id":<String_value>}]}
```

mpssession

Configuration for Client Session resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_external_auth	<Boolean>	Read-write	Is session created by using external authentication.
id	<String>	Read-write	Id is system generated key for all the client sessions.
tenant_name	<String>	Read-only	Tenant Name.
expire_after	<Integer>	Read-only	Session will expire after these many seconds.
login_time	<Integer>	Read-only	Session was initiated at this time.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
permission	<String>	Read-only	Permission to identify who created the session.
session_timeout	<Integer>	Read-only	Session Timeout set for this session.
port	<Integer>	Read-only	Port from where this session was initiated.
username	<String>	Read-only	User Name who initiated this session.
client_type	<String>	Read-only	Client Type.
last_activity_time	<Integer>	Read-only	Last Activity Time for this session.
ip_address	<String>	Read-only	IP Aaddress from where this session was initiated.
is_self	<Boolean>	Read-only	true, if this session is for current logged-in user.

Operations

(click to see Properties)

- GET (ALL)
- GET
- DELETE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/mpssession **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/mpssession **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of mpssession resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/mpssession **? pagesize=#no;pageno=#no** Use this query-parameter to get the mpssession resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/mpssession **?count=yes** Use this query-parameter to get the count of mpssession resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mpssession":[{

"tenant_name":<String_value>,

"expire_after":<Integer_value>,

"login_time":<Integer_value>,

"permission":<String_value>,

"session_timeout":<Integer_value>,

"port":<Integer_value>,

"username":<String_value>,

"client_type":<String_value>,

"last_activity_time":<Integer_value>,

"session_creator":<String_value>,
```

```
"is_external_auth":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"token":<String_value>,

"sessionid":<String_value>,

"is_self":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/mpssession/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mpssession":[{

"tenant_name":<String_value>,

"expire_after":<Integer_value>,
```

```
"login_time":<Integer_value>,

"permission":<String_value>,

"session_timeout":<Integer_value>,

"port":<Integer_value>,

"username":<String_value>,

"client_type":<String_value>,

"last_activity_time":<Integer_value>,

"session_creator":<String_value>,

"is_external_auth":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"token":<String_value>,

"sessionid":<String_value>,

"is_self":<Boolean_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/mpssession/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

mpsuser

Configuration for System User resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
session_timeout	<Integer>	Read-write	Session timeout for the user.
name	<String>	Read-write	User Name. Minimum length = 1 Maximum length = 128
session_timeout_unit	<String>	Read-write	Session timeout unit for the user.
external_authentication	<Boolean>	Read-write	Enable external authentication.
enable_session_timeout	<Boolean>	Read-write	Enables session timeout for user.
tenant_id	<String>	Read-write	Tenant Id of the system users. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
password	<String>	Read-write	Password. Minimum length = 1 Maximum length = 128
id	<String>	Read-write	Id is system generated key for all the system users.
unlock	<Boolean>	Read-write	A flag to unlock the user.
groups	<String[]>	Read-write	Groups to which user belongs.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/mpsuser?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"mpsuser": {

<b>"name":<String_value></b>,
```

```
<b>"password":<String_value></b>,

"session_timeout":<Integer_value>,

"session_timeout_unit":<String_value>,

"external_authentication":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"tenant_id":<String_value>,

"groups":<String_value[]>,

"id":<String_value>,

"unlock":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mpsuser":[{

"is_default":<Boolean_value>,

"session_timeout":<Integer_value>,

"permission":<String_value>,

"name":<String_value>,

"session_timeout_unit":<String_value>,

"external_authentication":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,
```

```
"tenant_id":<String_value>,

"password":<String_value>,

"one_way_password":<String_value>,

"id":<String_value>,

"one_way_pbkdf2_password":<String_value>,

"tenant_name":<String_value>,

"encrypted":<Boolean_value>,

"unlock":<Boolean_value>,

"issuperuser":<Boolean_value>,

"force_change_password":<Boolean_value>,

"groups":<String_value>,

"nsroot_user":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/mpsuser/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/mpsuser **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/mpsuser **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of mpsuser resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/mpsuser **? pagesize=#no;pageno=#no** Use this query-parameter to get the mpsuser resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/mpsuser **?count=yes** Use this query-parameter to get the count of mpsuser resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mpsuser":[{

"is_default":<Boolean_value>,
```

```
"session_timeout":<Integer_value>,

"permission":<String_value>,

"name":<String_value>,

"session_timeout_unit":<String_value>,

"external_authentication":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"tenant_id":<String_value>,

"password":<String_value>,

"one_way_password":<String_value>,

"id":<String_value>,

"one_way_pbkdf2_password":<String_value>,

"tenant_name":<String_value>,

"encrypted":<Boolean_value>,

"unlock":<Boolean_value>,

"issuperuser":<Boolean_value>,

"force_change_password":<Boolean_value>,

"groups":<String_value>,

"nsroot_user":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/mpsuser/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mpsuser":[{

"is_default":<Boolean_value>,

"session_timeout":<Integer_value>,

"permission":<String_value>,

"name":<String_value>,

"session_timeout_unit":<String_value>,

"external_authentication":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"tenant_id":<String_value>,

"password":<String_value>,

"one_way_password":<String_value>,

"id":<String_value>,

"one_way_pbkdf2_password":<String_value>,
```

```
"tenant_name":<String_value>,

"encrypted":<Boolean_value>,

"unlock":<Boolean_value>,

"issuperuser":<Boolean_value>,

"force_change_password":<Boolean_value>,

"groups":<String_value>,

"nsroot_user":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/mpsuser/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"mpsuser":{

<b>"id":<String_value></b>,

"session_timeout":<Integer_value>,

"session_timeout_unit":<String_value>,

"external_authentication":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,
```

```
"password":<String_value>,

"tenant_id":<String_value>,

"groups":<String_value[]>,

"unlock":<Boolean_value>,

"name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"mpsuser":[{

"is_default":<Boolean_value>,

"session_timeout":<Integer_value>,

"permission":<String_value>,

"name":<String_value>,

"session_timeout_unit":<String_value>,

"external_authentication":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"tenant_id":<String_value>,

"password":<String_value>,

"one_way_password":<String_value>,

"id":<String_value>,
```

```
"one_way_pbkdf2_password":<String_value>,

"tenant_name":<String_value>,

"encrypted":<Boolean_value>,

"unlock":<Boolean_value>,

"issuperuser":<Boolean_value>,

"force_change_password":<Boolean_value>,

"groups":<String_value>,

"nsroot_user":<String_value>}]}
```

ntp_param

Configuration for NTP Parameters resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
trusted_key_list	<Integer[]>	Read-write	List of Trusted Key Identifiers for Symmetric Key Cryptography. Minimum value = 1 Maximum value =
automax_logsec	<Integer>	Read-write	Automax Interval (as power of 2 in seconds).
revoke_logsec	<Integer>	Read-write	Revoke Interval (as power of 2 in seconds).

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
authentication	<Boolean>	Read-write	Authentication Enabled.

Operations

(click to see Properties)

- GET (ALL)
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ntp_param **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ntp_param **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ntp_param resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ntp_param **? pagesize=#no;pageno=#no** Use this query-parameter to get the ntp_param resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ntp_param **?count=yes** Use this query-parameter to get the count of ntp_param resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ntp_param":[{

"trusted_key_list":<Integer_value>,
```

```
"automax_logsec":<Integer_value>,

"revoke_logsec":<Integer_value>,

"authentication":<Boolean_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/ntp_param/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ntp_param":{

"trusted_key_list":<Integer_value[]>,

"automax_logsec":<Integer_value>,

"revoke_logsec":<Integer_value>,

"authentication":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ntp_param":[{

"trusted_key_list":<Integer_value>,

"automax_logsec":<Integer_value>,

"revoke_logsec":<Integer_value>,

"authentication":<Boolean_value>}]}
```

ntp_server

Configuration for NTP Server Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
preferred_server	<Boolean>	Read-write	NTP Server Preferred.
maxpoll	<Integer>	Read-write	Maximum Poll Interval. Maximum value =
minpoll	<Integer>	Read-write	Minimum Poll Interval. Maximum value =
autokey	<Boolean>	Read-write	Autokey Public Key Authentication.
key_id	<Integer>	Read-write	Key Identifier for Symmetric Key Authentication. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
server	<String>	Read-write	NTP Time Server Address.
client	<String>	Read-write	Sender of request, whether from Setup Wizard or direct NTP configuration.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/ntp_server?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ntp_server": {

<b>"server":<String_value></b>,

"client":<String_value>,

"minpoll":<Integer_value>,

"maxpoll":<Integer_value>,

"preferred_server":<Boolean_value>,

"autokey":<Boolean_value>,

"key_id":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ntp_server":[{

"preferred_server":<Boolean_value>,

"maxpoll":<Integer_value>,

"minpoll":<Integer_value>,

"autokey":<Boolean_value>,

"key_id":<Integer_value>,

"server":<String_value>,

"client":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/ntp_server/server_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ntp_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ntp_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ntp_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ntp_server **? pagesize=#no;pageno=#no** Use this query-parameter to get the ntp_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ntp_server **?count=yes** Use this query-parameter to get the count of ntp_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ntp_server":[{

"preferred_server":<Boolean_value>,

"maxpoll":<Integer_value>,

"minpoll":<Integer_value>,

"autokey":<Boolean_value>,

"key_id":<Integer_value>,

"server":<String_value>,

"client":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ntp_server/server_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ntp_server":[{

"preferred_server":<Boolean_value>,

"maxpoll":<Integer_value>,

"minpoll":<Integer_value>,

"autokey":<Boolean_value>,

"key_id":<Integer_value>,

"server":<String_value>,

"client":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/ntp_server/server_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ntp_server":{

"client":<String_value>,

"minpoll":<Integer_value>,

"maxpoll":<Integer_value>,

"preferred_server":<Boolean_value>,
```

```
"autokey":<Boolean_value>,

"key_id":<Integer_value>,

"server":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ntp_server":[{

"preferred_server":<Boolean_value>,

"maxpoll":<Integer_value>,

"minpoll":<Integer_value>,

"autokey":<Boolean_value>,

"key_id":<Integer_value>,

"server":<String_value>,

"client":<String_value>}]}
```

ntp_sync

Configuration for NTP Sync configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ntpd_status	<Boolean>	Read-write	ntpd status.

Operations

(click to see Properties)

- GET (ALL)
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ntp_sync **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ntp_sync **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ntp_sync resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ntp_sync **? pagesize=#no;pageno=#no** Use this query-parameter to get the ntp_sync resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ntp_sync **?count=yes** Use this query-parameter to get the count of ntp_sync resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ntp_sync":[{

"reboot":<Boolean_value>,

"ntpd_status":<Boolean_value>}]}
```

```

```

update

URL: https://<MGMT-IP>/nitro/v2/config/ntp_sync/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ntp_sync":{

"ntpd_status":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ntp_sync":[{

"reboot":<Boolean_value>,

"ntpd_status":<Boolean_value>}]}
```

password_policy

Configuration for Password Policy configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
enable_password_complexity	<Boolean>	Read-write	Enable user Password complexity.
minimum_password_length	<Integer>	Read-write	Minimum password length . Minimum value = 6 Maximum value =
id	<String>	Read-write	Id is system generated key.

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/password_policy **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/password_policy **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of password_policy resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/password_policy **? pagesize=#no;pageno=#no** Use this query-parameter to get the password_policy resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/password_policy **?count=yes** Use this query-parameter to get the count of password_policy resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"password_policy":[{

"enable_password_complexity":<Boolean_value>,

"minimum_password_length":<Integer_value>,

"id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/password_policy/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"password_policy":[{

"enable_password_complexity":<Boolean_value>,

"minimum_password_length":<Integer_value>,

"id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/password_policy/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"password_policy":{

"id":<String_value>,

"enable_password_complexity":<Boolean_value>,

"minimum_password_length":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"password_policy":[{

"enable_password_complexity":<Boolean_value>,

"minimum_password_length":<Integer_value>,

"id":<String_value>}]}
```

platform_specs

Configuration for Platform Specification resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cpu_cores_per_vm_list	<Integer[]>	Read-only	List of CPU cores which can be allocated to VM.
contains_fortville_nics	<Boolean>	Read-only	check if Fortville nic card is present.
max_channels	<Integer>	Read-only	Maximum number of channels available.
contains_cavium	<Boolean>	Read-only	check if Cavium chips are present.
no_of_mgmt_interfaces	<Integer>	Read-only	total number of management interfaces available.
contains_raid	<Boolean>	Read-only	Enable RAID for the platform.
cavium_vf_to_acu_x	<Integer>	Read-only	Maximum number of CPU cores which

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			can be allocated to VM.
contains_swraid	<Boolean>	Read-only	Enable SW RAID for the platform.
is_fips_present	<Boolean>	Read-only	To check if FIPS is present.
is_nextgen	<Boolean>	Read-only	If NextGen Platform.
max_cores_per_vm	<Integer>	Read-only	Maximum number of CPU cores which can be allocated to VM.
cavium_vf_to_scu_x	<Integer>	Read-only	Maximum number of CPU cores which can be allocated to VM.
contains_mnx_nics	<Boolean>	Read-only	check if mellanox nics card are present.
contains_coleto	<Boolean>	Read-only	check if Coleto chips are present.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/platform_specs **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/platform_specs **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of platform_specs resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/platform_specs **? pagesize=#no;pageno=#no** Use this query-parameter to get the platform_specs resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/platform_specs **?count=yes** Use this query-parameter to get the count of platform_specs resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"platform_specs":[{

"cpu_cores_per_vm_list":<Integer_value>,

"contains_fortville_nics":<Boolean_value>,

"max_channels":<Integer_value>,

"contains_cavium":<Boolean_value>,

"no_of_mgmt_interfaces":<Integer_value>,

"contains_raid":<Boolean_value>,

"cavium_vf_to_acu_x":<Integer_value>,

"contains_swraid":<Boolean_value>,

"is_fips_present":<Boolean_value>,

"is_nextgen":<Boolean_value>,
```

```
"max_cores_per_vm":<Integer_value>,

"cavium_vf_to_scu_x":<Integer_value>,

"contains_mnx_nics":<Boolean_value>,

"contains_coleto":<Boolean_value>}]}
```

property_map

Configuration for server properties resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
prop_value	<String>	Read-write	Property value. Minimum length = 1 Maximum length = 512
prop_key	<String>	Read-write	Property key. Minimum length = 1 Maximum length = 128

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

prune_policy

Configuration for Prune Policy resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
do_auto_purge	<Boolean>	Read-write	True, if user wants to opt for auto DB data purging once disk usage reaches threshold value.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
policy_name	<String>	Read-write	Policy Name. Minimum length = 1 Maximum length = 128
data_to_keep_in_days	<Integer>	Read-write	Number of days data to retain. Minimum value = 1 Maximum value =
disk_prune_threshold	<Integer>	Read-write	Disk Pruning Threshold. Minimum value = 10 Maximum value =

Operations

(click to see Properties)

- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get

URL: https://<MGMT-IP>/nitro/v2/config/prune_policy **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"prune_policy":[{

"do_auto_purge":<Boolean_value>,

"policy_name":<String_value>,

"data_to_keep_in_days":<Integer_value>,

"disk_prune_threshold":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/prune_policy/policy_name_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"prune_policy":{

<b>"data_to_keep_in_days":<Integer_value></b>,

<b>"disk_prune_threshold":<Integer_value></b>,

"policy_name":<String_value>,

"do_auto_purge":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"prune_policy":[{

"do_auto_purge":<Boolean_value>,

"policy_name":<String_value>,

"data_to_keep_in_days":<Integer_value>,

"disk_prune_threshold":<Integer_value>}]}
```

radius_server

Configuration for Radius Server configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
auth_timeout	<Integer>	Read-write	The maximum number of seconds the system will wait for a response from the Radius server. Maximum value =
group_separator	<String>	Read-write	Group separator string that delimits group names within a RADIUS attribute for RADIUS group extraction. Maximum length = 7
pass_encoding	<String>	Read-write	Enable password encoding in RADIUS

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			packets send to the RADIUS server.
enable_nas_ip	<Boolean>	Read-write	Enable NAS IP extraction.
radius_key	<String>	Read-write	Key of radius server. Minimum length = 4 Maximum length = 32
default_authentication_group	<String>	Read-write	This is the default group that is chosen when the authentication succeeds in addition to extracted groups. Maximum length = 64
ip_vendor_id	<Integer>	Read-write	The vendor ID of the attribute in the RADIUS response which denotes the intranet IP. Maximum value =
id	<String>	Read-write	Id is system generated key for all the radius servers.
ip_address	<String>	Read-write	IP Address of radius server. Minimum length = 1 Maximum length = 64
ip_attribute_type	<Integer>	Read-write	The attribute type of the remote IP

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			address attribute in a RADIUS response. Maximum value =
attribute_type	<Integer>	Read-write	Attribute type for RADIUS group extraction. Maximum value =
nas_id	<String>	Read-write	NAS ID. Maximum length = 128
vendor_id	<Integer>	Read-write	Vendor ID for RADIUS group extraction. Maximum value =
name	<String>	Read-write	Name of radius server. Minimum length = 1 Maximum length = 128
port	<Integer>	Read-write	Port number of radius server. Minimum value = 1 Maximum value =
groups_prefix	<String>	Read-write	Prefix string that precedes group names within a RADIUS attribute for RADIUS group extraction. Maximum length = 31

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
pwd_vendor_id	<Integer>	Read-write	Vendor ID of the password in the RADIUS response. Used to extract the user password. Maximum value =
pwd_attribute_type	<Integer>	Read-write	The attribute type of the password attribute in a RADIUS response.. Maximum value =
accounting	<Boolean>	Read-write	Enable accounting in the radius server.
address_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, -1: Hostname.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/radius_server?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"radius_server": {

<b>"radius_key":<String_value></b>,
```

```
<b>"ip_address":<String_value></b>,

<b>"name":<String_value></b>,

"auth_timeout":<Integer_value>,

"group_separator":<String_value>,

"pass_encoding":<String_value>,

"enable_nas_ip":<Boolean_value>,

"default_authentication_group":<String_value>,

"id":<String_value>,

"ip_attribute_type":<Integer_value>,

"pwd_vendor_id":<Integer_value>,

"groups_prefix":<String_value>,

"port":<Integer_value>,

"pwd_attribute_type":<Integer_value>,

"accounting":<Boolean_value>,

"address_type":<Integer_value>,

"ip_vendor_id":<Integer_value>,

"attribute_type":<Integer_value>,

"nas_id":<String_value>,

"vendor_id":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"radius_server":[{

"auth_timeout":<Integer_value>,

"group_separator":<String_value>,

"pass_encoding":<String_value>,

"enable_nas_ip":<Boolean_value>,

"radius_key":<String_value>,

"default_authentication_group":<String_value>,

"ip_vendor_id":<Integer_value>,

"id":<String_value>,

"ip_address":<String_value>,

"ip_attribute_type":<Integer_value>,

"attribute_type":<Integer_value>,

"nas_id":<String_value>,

"vendor_id":<Integer_value>,

"name":<String_value>,

"port":<Integer_value>,

"groups_prefix":<String_value>,

"pwd_vendor_id":<Integer_value>,

"pwd_attribute_type":<Integer_value>,

"accounting":<Boolean_value>,
```

```
"address_type":<Integer_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/radius_server/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/radius_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/radius_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of radius_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/radius_server **? pagesize=#no;pageno=#no** Use this query-parameter to get the radius_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/radius_server **?count=yes** Use this query-parameter to get the count of radius_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"radius_server":[{

"auth_timeout":<Integer_value>,

"group_separator":<String_value>,

"pass_encoding":<String_value>,

"enable_nas_ip":<Boolean_value>,

"radius_key":<String_value>,

"default_authentication_group":<String_value>,

"ip_vendor_id":<Integer_value>,

"id":<String_value>,

"ip_address":<String_value>,

"ip_attribute_type":<Integer_value>,

"attribute_type":<Integer_value>,

"nas_id":<String_value>,
```

```
"vendor_id":<Integer_value>,

"name":<String_value>,

"port":<Integer_value>,

"groups_prefix":<String_value>,

"pwd_vendor_id":<Integer_value>,

"pwd_attribute_type":<Integer_value>,

"accounting":<Boolean_value>,

"address_type":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/radius_server/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"radius_server":[{

"auth_timeout":<Integer_value>,

"group_separator":<String_value>,

"pass_encoding":<String_value>,

"enable_nas_ip":<Boolean_value>,

"radius_key":<String_value>,

"default_authentication_group":<String_value>,

"ip_vendor_id":<Integer_value>,

"id":<String_value>,

"ip_address":<String_value>,

"ip_attribute_type":<Integer_value>,

"attribute_type":<Integer_value>,

"nas_id":<String_value>,

"vendor_id":<Integer_value>,

"name":<String_value>,

"port":<Integer_value>,

"groups_prefix":<String_value>,

"pwd_vendor_id":<Integer_value>,

"pwd_attribute_type":<Integer_value>,

"accounting":<Boolean_value>,

"address_type":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/radius_server/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"radius_server":{

<b>"id":<String_value></b>,

<b>"ip_address":<String_value></b>,

<b>"name":<String_value></b>,

"auth_timeout":<Integer_value>,

"group_separator":<String_value>,

"pass_encoding":<String_value>,

"enable_nas_ip":<Boolean_value>,

"default_authentication_group":<String_value>,

"ip_attribute_type":<Integer_value>,

"pwd_vendor_id":<Integer_value>,

"groups_prefix":<String_value>,

"port":<Integer_value>,
```

```
"pwd_attribute_type":<Integer_value>,

"accounting":<Boolean_value>,

"address_type":<Integer_value>,

"radius_key":<String_value>,

"ip_vendor_id":<Integer_value>,

"attribute_type":<Integer_value>,

"nas_id":<String_value>,

"vendor_id":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"radius_server":[{

"auth_timeout":<Integer_value>,

"group_separator":<String_value>,

"pass_encoding":<String_value>,

"enable_nas_ip":<Boolean_value>,

"radius_key":<String_value>,

"default_authentication_group":<String_value>,

"ip_vendor_id":<Integer_value>,

"id":<String_value>,
```

```
"ip_address":<String_value>,

"ip_attribute_type":<Integer_value>,

"attribute_type":<Integer_value>,

"nas_id":<String_value>,

"vendor_id":<Integer_value>,

"name":<String_value>,

"port":<Integer_value>,

"groups_prefix":<String_value>,

"pwd_vendor_id":<Integer_value>,

"pwd_attribute_type":<Integer_value>,

"accounting":<Boolean_value>,

"address_type":<Integer_value>}]}
```

report_timezone

Configuration for set report_timezone local or gmt. resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
timezone	<String>	Read-write	Timezone local or gmt.
tz_offset	<Integer>	Read-write	Timezone offset.

Operations

(click to see Properties)

- GET (ALL)
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/report_timezone **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/report_timezone **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of report_timezone resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/report_timezone **?pagesize=#no;pageno=#no** Use this query-parameter to get the report_timezone resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/report_timezone **?count=yes** Use this query-parameter to get the count of report_timezone resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"report_timezone":[{

"timezone":<String_value>,

"tz_offset":<Integer_value>}]}
```

modify

URL: https://<MGMT-IP>/nitro/v2/config/report_timezone/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"report_timezone":{

"timezone":<String_value>,

"tz_offset":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"report_timezone":[{

"timezone":<String_value>,

"tz_offset":<Integer_value>}]}
```

restore

Configuration for Restore resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_valid_license	<Boolean>	Read-write	To indicate if available license is valid and (NS+throughput) sufficient for restore..
backup_components	<backup_component []>	Read-write	Backup Components.
required_throughput	<Double>	Read-write	Throughput required by the backedup instances to get restored..
max_allowed_ns	<Integer>	Read-write	Number of Citrix ADC instances current license allows to be restored..
restore_type	<Integer>	Read-write	.
backup_password	<String>	Read-write	Password of encrypted backup file.
required_ns	<Integer>	Read-write	Total number of instances present in backup file..
ip_address_list	<String[]>	Read-write	List of VM IP Address. Minimum length = 1 Maximum length = 1024
sbi_file	<String>	Read-write	SDX Image selected by User for Restore.
backup_file_name	<String>	Read-write	Backup file name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
max_allowed_throughput	<Double>	Read-write	Maximum throughput current license allows..
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- ADD
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/restore?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"restore": {

"backup_components":[{

"component_message":<String_value>,

"component_hostname":<String_value>,

"component_version_bit":<String_value>,

"component_error_severity":<Integer_value>,

"component_versions":<String_value>,

"component_minimum_version":<String_value>,

"component_name":<String_value>,

"component":<String_value>,
```

```
"backup_file_name":<String_value>,

"component_ip":<String_value>}],

"backup_file_name":<String_value>,

"max_allowed_throughput":<Double_value>,

"is_valid_license":<Boolean_value>,

"restore_type":<Integer_value>,

"max_allowed_ns":<Integer_value>,

"backup_password":<String_value>,

"ip_address_list":<String_value[]>,

"required_ns":<Integer_value>,

"sbi_file":<String_value>,

"required_throughput":<Double_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"restore":[{

"is_valid_license":<Boolean_value>,

"backup_components":[{

"component_message":<String_value>,

"component_versions":<String_value>,
```

```
"component_minimum_version":<String_value>,

"backup_file_name":<String_value>,

"component":<String_value>,

"component_name":<String_value>,

"component_ip":<String_value>,

"component_version_bit":<String_value>,

"component_hostname":<String_value>,

"component_error_severity":<Integer_value>}],

"required_throughput":<Double_value>,

"max_allowed_ns":<Integer_value>,

"restore_type":<Integer_value>,

"backup_password":<String_value>,

"required_ns":<Integer_value>,

"ip_address_list":<String_value>,

"sbi_file":<String_value>,

"backup_file_name":<String_value>,

"act_id":<String_value>,

"max_allowed_throughput":<Double_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/restore **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/restore **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of restore resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/restore **? pagesize=#no;pageno=#no** Use this query-parameter to get the restore resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/restore **?count=yes** Use this query-parameter to get the count of restore resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"restore":[{

"is_valid_license":<Boolean_value>,

"backup_components":[{

"component_message":<String_value>,

"component_versions":<String_value>,

"component_minimum_version":<String_value>,
```

```
"backup_file_name":<String_value>,

"component":<String_value>,

"component_name":<String_value>,

"component_ip":<String_value>,

"component_version_bit":<String_value>,

"component_hostname":<String_value>,

"component_error_severity":<Integer_value>}],

"required_throughput":<Double_value>,

"max_allowed_ns":<Integer_value>,

"restore_type":<Integer_value>,

"backup_password":<String_value>,

"required_ns":<Integer_value>,

"ip_address_list":<String_value>,

"sbi_file":<String_value>,

"backup_file_name":<String_value>,

"act_id":<String_value>,

"max_allowed_throughput":<Double_value>}]}
```

sb_info

Configuration for Single Bundle File Info resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
component_info	<String[]>	Read-only	Component Info.
error_message	<String>	Read-only	Error Message in case the Single Bundle cannot be applied.
total_approx_time	<Integer>	Read-only	Total approximate time that the upgrade will take.
reboot_message	<String>	Read-only	Reboot Message.
selected_sb_version	<String>	Read-only	Selected Single Bundle Version.
current_sb_version	<String>	Read-only	Current Single Bundle Version.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/sb_info **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/sb_info **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of sb_info resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/sb_info **? pagesize=#no;pageno=#no** Use this query-parameter to get the sb_info resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/sb_info **?count=yes** Use this query-parameter to get the count of sb_info resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sb_info":[{

"component_info":<String_value>,

"error_message":<String_value>,

"file_name":<String_value>,

"total_approx_time":<Integer_value>,

"reboot_message":<String_value>,

"selected_sb_version":<String_value>,

"current_sb_version":<String_value>}]}
```

sdx_autoreg

Configuration for SDX auto registration control information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
autoreg	<Boolean>	Read-write	Flag to control autoreg.
is_banner_displayed	<Boolean>	Read-write	Flag to indicate whether banner has been displayed.
id	<String>	Read-write	Id is system generated key.

Operations

(click to see Properties)

- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get

URL: https://<MGMT-IP>/nitro/v2/config/sdx_autoreg **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_autoreg":[{

"autoreg":<Boolean_value>,

"is_banner_displayed":<Boolean_value>,

"id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/sdx_autoreg/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sdx_autoreg":{

"autoreg":<Boolean_value>,

"is_banner_displayed":<Boolean_value>,

"id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_autoreg":[{

"autoreg":<Boolean_value>,

"is_banner_displayed":<Boolean_value>,

"id":<String_value>}]}
```

sdx_backup_restore

Configuration for Backup resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ip_address_list	<String[]>	Read-write	List of VM IP Address. Minimum length = 1 Maximum length = 1024
reset_type	<Integer>	Read-write	Reset Type [0: Reset (Without Network Configuration), 1: Reset (With Network Configuration), 2: Appliance Reset, 3: Appliance Restore, 4: Instance Restore, 5: Backup].
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
name	<String>	Read-only	Name of VM device.
act_id	<String>	Read-only	Activity Id.
ip_address	<String>	Read-only	IP Address for this VM device.
type	<String>	Read-only	Type of device, (br | ns).

Operations

(click to see Properties)

- FACTORYRESET
- GET
- RESTORE
- BACKUP

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

factoryreset

URL: https://<MGMT-IP>/nitro/v2/config/sdx_backup_restore?action=factoryreset;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sdx_backup_restore": {

"file_name":<String_value>,

"reset_type":<Integer_value>,

"ip_address_list":<String_value[]>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_backup_restore":[{

"restore_type":<Integer_value>,

"ip_address_list":<String_value>,

"name":<String_value>,

"reset_type":<Integer_value>,

"sync_operation":<Boolean_value>,
```

```
"file_name":<String_value>,

"act_id":<String_value>,

"ip_address":<String_value>,

"type":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/sdx_backup_restore **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_backup_restore":[{

"restore_type":<Integer_value>,

"ip_address_list":<String_value>,

"name":<String_value>,

"reset_type":<Integer_value>,
```

```
"sync_operation":<Boolean_value>,

"file_name":<String_value>,

"act_id":<String_value>,

"ip_address":<String_value>,

"type":<String_value>}]}
```

restore

URL: https://<MGMT-IP>/nitro/v2/config/sdx_backup_restore?action=restore;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sdx_backup_restore": {

"file_name":<String_value>,

"reset_type":<Integer_value>,

"ip_address_list":<String_value[]>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_backup_restore":[{

"restore_type":<Integer_value>,

"ip_address_list":<String_value>,

"name":<String_value>,

"reset_type":<Integer_value>,

"sync_operation":<Boolean_value>,

"file_name":<String_value>,

"act_id":<String_value>,

"ip_address":<String_value>,

"type":<String_value>}]}
```

backup

URL: https://<MGMT-IP>/nitro/v2/config/sdx_backup_restore?action=backup;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sdx_backup_restore": {

"file_name":<String_value>,

"reset_type":<Integer_value>,
```

```
"ip_address_list":<String_value[]>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_backup_restore":[{

"restore_type":<Integer_value>,

"ip_address_list":<String_value>,

"name":<String_value>,

"reset_type":<Integer_value>,

"sync_operation":<Boolean_value>,

"file_name":<String_value>,

"act_id":<String_value>,

"ip_address":<String_value>,

"type":<String_value>}]}
```

sdx_builtin_agent

Configuration for SDX built in agent registration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
activation_code	<String>	Read-write	activation code.
service_url	<String>	Read-write	service url.
vpx_user_name	<String>	Read-write	vpx user name.
status	<String>	Read-write	builtin agent conn status.
sdx_user_name	<String>	Read-write	sdx user name.
vpx_password	<String>	Read-write	vpx password.
sdx_password	<String>	Read-write	sdx password.
profile_name	<String>	Read-write	sdx device profile name.

Operations

(click to see Properties)

- REGISTER
- GET (ALL)
- DE_REGISTER

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

register

URL: https://<MGMT-IP>/nitro/v2/config/sdx_builtin_agent?action=register;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sdx_builtin_agent": {

"vpx_password":<String_value>,

"activation_code":<String_value>,

"vpx_user_name":<String_value>,
```

```
"sdx_password":<String_value>,

"status":<String_value>,

"sdx_user_name":<String_value>,

"service_url":<String_value>,

"profile_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_builtin_agent":[{

"activation_code":<String_value>,

"service_url":<String_value>,

"vpx_user_name":<String_value>,

"status":<String_value>,

"sdx_user_name":<String_value>,

"vpx_password":<String_value>,

"sdx_password":<String_value>,

"profile_name":<String_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/sdx_builtin_agent **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/sdx_builtin_agent **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of sdx_builtin_agent resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/sdx_builtin_agent **?pagesize=#no;pageno=#no** Use this query-parameter to get the sdx_builtin_agent resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/sdx_builtin_agent **?count=yes** Use this query-parameter to get the count of sdx_builtin_agent resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_builtin_agent":[{

"activation_code":<String_value>,

"service_url":<String_value>,

"vpx_user_name":<String_value>,

"status":<String_value>,
```

```
"sdx_user_name":<String_value>,

"vpx_password":<String_value>,

"sdx_password":<String_value>,

"profile_name":<String_value>}]}
```

de_register

URL: https://<MGMT-IP>/nitro/v2/config/sdx_builtin_agent?action=de_register;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sdx_builtin_agent": {

"vpx_password":<String_value>,

"activation_code":<String_value>,

"vpx_user_name":<String_value>,

"sdx_password":<String_value>,

"status":<String_value>,

"sdx_user_name":<String_value>,

"service_url":<String_value>,

"profile_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_builtin_agent":[{

"activation_code":<String_value>,

"service_url":<String_value>,

"vpx_user_name":<String_value>,

"status":<String_value>,

"sdx_user_name":<String_value>,

"vpx_password":<String_value>,

"sdx_password":<String_value>,

"profile_name":<String_value>}]}
```

sdx_network_config

Configuration for SDX Network Configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
dns	<String>	Read-write	DNS Server. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
gateway	<String>	Read-write	Gateway . Minimum length = 1 Maximum length = 64
svm_ipv6_address	<String>	Read-write	Management Service IPv6 Address.
additional_dns2	<String>	Read-write	Additional DNS Server. Can be IPv4 or IPv6.
additional_dns1	<String>	Read-write	Additional DNS Server. Can be IPv4 or IPv6.
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
svm_ip_address	<String>	Read-write	Management Service IP Address. Minimum length = 1 Maximum length = 64
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
netmask	<String>	Read-write	Netmask. Minimum length = 1 Maximum length = 64
xen_netmask	<String>	Read-write	Xen Netmask. Minimum length = 1 Maximum length = 64
xen_gateway	<String>	Read-write	XEN Gateway. Minimum length = 1 Maximum length = 64
xen_dns	<String>	Read-write	Xen Netmask. Minimum length = 1 Maximum length = 64
dns_v6	<String>	Read-write	DNS Server IPv6 Address.
xen_ip_address	<String>	Read-write	XenServer IP Address. Minimum length = 1 Maximum length = 64
apply_svm_ip	<Boolean>	Read-write	Apply SVM IP.
network_interface	<String>	Read-write	Interface on which management needs to be enabled. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 15
is_xen_reachable	<Boolean>	Read-write	Flag to set/unset external access to Xen.
sync_to_vm	<Boolean>	Read-write	Set option to sync or not to sync with CB.
skip_reboot	<Boolean>	Read-write	Set Citrix SD-WAN reboot required option during PUT.
init_status	<Integer>	Read-write	System initialize status.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get

URL: https://<MGMT-IP>/nitro/v2/config/sdx_network_config **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_network_config":[{

"dns":<String_value>,

"gateway":<String_value>,

"actual_network_interface":<String_value>,

"svm_ipv6_address":<String_value>,

"additional_dns2":<String_value>,

"additional_dns1":<String_value>,

"config_type":<Integer_value>,

"svm_ip_address":<String_value>,

"gateway_ipv6":<String_value>,

"host_ip_address":<String_value>,

"netmask":<String_value>,

"xen_netmask":<String_value>,

"xen_gateway":<String_value>,

"xen_dns":<String_value>,

"dns_v6":<String_value>,

"xen_ip_address":<String_value>,

"apply_svm_ip":<Boolean_value>,

"network_interface":<String_value>,

"is_xen_reachable":<Boolean_value>,

"sync_to_vm":<Boolean_value>,
```

```
"skip_reboot":<Boolean_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>,

"init_status":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/sdx_network_config/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sdx_network_config":{

<b>"gateway":<String_value></b>,

<b>"config_type":<Integer_value></b>,

<b>"netmask":<String_value></b>,

<b>"xen_netmask":<String_value></b>,

<b>"xen_gateway":<String_value></b>,

<b>"xen_ip_address":<String_value></b>,

<b>"network_interface":<String_value></b>,

"skip_reboot":<Boolean_value>,
```

```
"sync_to_vm":<Boolean_value>,

"dns":<String_value>,

"init_status":<Integer_value>,

"xen_dns":<String_value>,

"svm_ipv6_address":<String_value>,

"svm_ip_address":<String_value>,

"additional_dns1":<String_value>,

"additional_dns2":<String_value>,

"dns_v6":<String_value>,

"gateway_ipv6":<String_value>,

"apply_svm_ip":<Boolean_value>,

"host_ip_address":<String_value>,

"is_xen_reachable":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdx_network_config":[{

"dns":<String_value>,

"gateway":<String_value>,

"actual_network_interface":<String_value>,
```

```
"svm_ipv6_address":<String_value>,

"additional_dns2":<String_value>,

"additional_dns1":<String_value>,

"config_type":<Integer_value>,

"svm_ip_address":<String_value>,

"gateway_ipv6":<String_value>,

"host_ip_address":<String_value>,

"netmask":<String_value>,

"xen_netmask":<String_value>,

"xen_gateway":<String_value>,

"xen_dns":<String_value>,

"dns_v6":<String_value>,

"xen_ip_address":<String_value>,

"apply_svm_ip":<Boolean_value>,

"network_interface":<String_value>,

"is_xen_reachable":<Boolean_value>,

"sync_to_vm":<Boolean_value>,

"skip_reboot":<Boolean_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>,

"init_status":<Integer_value>}]}
```

sdxtools_image

Configuration for SDXTools Build File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/sdxtools_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/sdxtools_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/sdxtools_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of sdxtools_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/sdxtools_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the sdxtools_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/sdxtools_image **?count=yes** Use this query-parameter to get the count of sdxtools_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdxtools_image":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/sdxtools_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdxtools_image":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/sdxtools_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

sdxtools_upgrade

Configuration for Upgrade SDXTools of GuestVM - Applicable only for GuestVM's that support SDXTools resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	SDXTools Image name. Minimum length = 1 Maximum length = 128
ip_address_arr	<String[]>	Read-write	List of GuestVM IP Address. Minimum length = 1 Maximum length = 64

Operations

(click to see Properties)

- UPGRADE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

upgrade

URL: https://<MGMT-IP>/nitro/v2/config/sdxtools_upgrade?action=upgrade;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sdxtools_upgrade": {

<b>"image_name":<String_value></b>,

<b>"ip_address_arr":<String_value[]></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sdxtools_upgrade":[{

"image_name":<String_value>,

"ip_address_arr":<String_value>}]}
```

sent_mails

Configuration for Sent mail record item keeping. resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
to_list	<String>	Read-write	List of to whom send the mail. .
failed_message	<String>	Read-write	Subject of the mail sent.. Minimum length = 1 Maximum length = 2000
subject	<String>	Read-write	Subject of the mail sent.. Minimum length = 1 Maximum length = 2000

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
bcc_list	<String>	Read-write	List to whom BCC the mail..
is_auth	<Boolean>	Read-write	Is authentication enabled for this smtp server.
cc_list	<String>	Read-write	List to whom CC the mail..
port	<Integer>	Read-write	SMTP Server port address.. Maximum value =
is_sent	<Boolean>	Read-write	Is this mail sent successfully..
message	<String>	Read-write	Content of the mail sent.. Minimum length = 1 Maximum length = 20000
username	<String>	Read-write	Username for the smtp server. Maximum length = 128
id	<String>	Read-write	Id is system generated key for all the sent mail record..
is_ssl	<Boolean>	Read-write	Is this smtp server is SSL support configured..
server_name	<String>	Read-write	SMTP server name. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
profile_name	<String>	Read-write	Profile name through which mail sent.. Minimum length = 1 Maximum length = 128
timestamp	<Double>	Read-only	Time when the mail was sent.

Operations

(click to see Properties)

- DELETE
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/sent_mails/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/sent_mails **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/sent_mails **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of sent_mails resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/sent_mails **?pagesize=#no;pageno=#no** Use this query-parameter to get the sent_mails resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/sent_mails **?count=yes** Use this query-parameter to get the count of sent_mails resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sent_mails":[{

"to_list":<String_value>,

"failed_message":<String_value>,

"subject":<String_value>,

"bcc_list":<String_value>,

"is_auth":<Boolean_value>,

"cc_list":<String_value>,

"port":<Integer_value>,
```

```
"is_sent":<Boolean_value>,

"message":<String_value>,

"username":<String_value>,

"timestamp":<Double_value>,

"id":<String_value>,

"is_ssl":<Boolean_value>,

"server_name":<String_value>,

"profile_name":<String_value>}]}
```

sent_sms

Configuration for Sent sms record item keeping. resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
to_list	<String>	Read-write	List of to whom send the sms. .
failed_message	<String>	Read-write	Subject of the sms sent.. Minimum length = 1 Maximum length = 2000
is_sent	<Boolean>	Read-write	Is this sms sent successfully..
message	<String>	Read-write	Content of the sms sent..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 20000
username	<String>	Read-write	Username for the sms server. Maximum length = 128
url	<String>	Read-write	URL used for sending the sms. .
id	<String>	Read-write	Id is system generated key for all the sent sms record..
is_ssl	<Boolean>	Read-write	Is this sms sent as SSL..
server_name	<String>	Read-write	SMS server name. Minimum length = 1 Maximum length = 128
profile_name	<String>	Read-write	Profile name through which sms sent.. Minimum length = 1 Maximum length = 128
timestamp	<Double>	Read-only	Time when the sms was sent.

Operations

(click to see Properties)

- DELETE

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/sent_sms/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/sent_sms **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/sent_sms **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of sent_sms resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/sent_sms **?pagesize=#no;pageno=#no** Use this query-parameter to get the sent_sms resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/sent_sms **?count=yes** Use this query-parameter to get the count of sent_sms resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sent_sms":[{

"to_list":<String_value>,

"failed_message":<String_value>,

"is_sent":<Boolean_value>,

"message":<String_value>,

"username":<String_value>,

"timestamp":<Double_value>,

"url":<String_value>,

"id":<String_value>,

"is_ssl":<Boolean_value>,

"server_name":<String_value>,

"profile_name":<String_value>}]}
```

server_connection

Configuration for server properties resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
server_rtt	<Integer>	Read-write	client RTT..
count	<Integer>	Read-write	count.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
longitude	<Double>	Read-write	longitude.
latitude	<Double>	Read-write	latitude.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

server_nodes

Configuration for Additional Server node information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
priority	<Integer>	Read-write	priority of the app node.
deployment_status	<Boolean>	Read-write	Deployment status of this node..
app_id	<String>	Read-write	Node ID of the server node.
name	<String>	Read-write	Server node IP Address.
db_health	<String>	Read-write	Database health of the node.
state	<String>	Read-write	State of the App node.
username	<String>	Read-write	username configured on Server node.. Minimum length = 1 Maximum length = 128
password	<String>	Read-write	Password. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
sync_progress	<String>	Read-write	Database synchronization status with the primary node.
is_pooled_license	<Boolean>	Read-write	is_pooled_license present on node.
is_primary	<Boolean>	Read-write	true if this node is primary.
is_star_node	<Boolean>	Read-write	This node is star node or not.
build_name	<String>	Read-write	Build name of the server node.
public_key	<String>	Read-write	Useful for sending the public key while registration.
device_count	<Integer>	Read-write	No of managed devices owned by the node.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/server_nodes?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"server_nodes": {

<b>"password":<String_value></b>,

"priority":<Integer_value>,

"deployment_status":<Boolean_value>,

"state":<String_value>,

"public_key":<String_value>,

"name":<String_value>,

"db_health":<String_value>,

"device_count":<Integer_value>,

"username":<String_value>,

"build_name":<String_value>,

"is_pooled_license":<Boolean_value>,

"is_star_node":<Boolean_value>,

"app_id":<String_value>,

"sync_progress":<String_value>,

"is_primary":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"server_nodes":[{
```

```
"priority":<Integer_value>,

"deployment_status":<Boolean_value>,

"app_id":<String_value>,

"name":<String_value>,

"db_health":<String_value>,

"state":<String_value>,

"username":<String_value>,

"password":<String_value>,

"sync_progress":<String_value>,

"is_pooled_license":<Boolean_value>,

"is_primary":<Boolean_value>,

"is_star_node":<Boolean_value>,

"build_name":<String_value>,

"public_key":<String_value>,

"device_count":<Integer_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/server_nodes/name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/server_nodes **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/server_nodes **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of server_nodes resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/server_nodes **?pagesize=#no;pageno=#no** Use this query-parameter to get the server_nodes resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/server_nodes **?count=yes** Use this query-parameter to get the count of server_nodes resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"server_nodes":[{

"priority":<Integer_value>,

"deployment_status":<Boolean_value>,

"app_id":<String_value>,

"name":<String_value>,

"db_health":<String_value>,

"state":<String_value>,

"username":<String_value>,

"password":<String_value>,

"sync_progress":<String_value>,

"is_pooled_license":<Boolean_value>,

"is_primary":<Boolean_value>,

"is_star_node":<Boolean_value>,

"build_name":<String_value>,

"public_key":<String_value>,

"device_count":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/server_nodes/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"server_nodes":[{

"priority":<Integer_value>,

"deployment_status":<Boolean_value>,

"app_id":<String_value>,

"name":<String_value>,

"db_health":<String_value>,

"state":<String_value>,

"username":<String_value>,

"password":<String_value>,

"sync_progress":<String_value>,

"is_pooled_license":<Boolean_value>,

"is_primary":<Boolean_value>,

"is_star_node":<Boolean_value>,

"build_name":<String_value>,
```

```
"public_key":<String_value>,

"device_count":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/server_nodes/name_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"server_nodes":{

<b>"name":<String_value></b>,

"priority":<Integer_value>,

"deployment_status":<Boolean_value>,

"state":<String_value>,

"password":<String_value>,

"public_key":<String_value>,

"db_health":<String_value>,

"device_count":<Integer_value>,

"username":<String_value>,

"build_name":<String_value>,
```

```
"is_pooled_license":<Boolean_value>,

"is_star_node":<Boolean_value>,

"app_id":<String_value>,

"sync_progress":<String_value>,

"is_primary":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"server_nodes":[{

"priority":<Integer_value>,

"deployment_status":<Boolean_value>,

"app_id":<String_value>,

"name":<String_value>,

"db_health":<String_value>,

"state":<String_value>,

"username":<String_value>,

"password":<String_value>,

"sync_progress":<String_value>,

"is_pooled_license":<Boolean_value>,

"is_primary":<Boolean_value>,
```

```
"is_star_node":<Boolean_value>,

"build_name":<String_value>,

"public_key":<String_value>,

"device_count":<Integer_value>}]}
```

service

Configuration for Service used in application container resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_name	<String>	Read-write	.
service_ip_address	<String>	Read-write	IPv4 or IPv6 Address assigned to the service. Minimum length = 1 Maximum length = 64
service_port	<Integer>	Read-write	Port number of the service.
parent_id	<String>	Read-write	.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

sms_profile

Configuration for SMS profile resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
to_list	<String>	Read-write	To list.. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the sms profile..
server_name	<String>	Read-write	SMS server name. Minimum length = 1 Maximum length = 128
profile_name	<String>	Read-write	Profile name for the sms setting.. Minimum length = 1 Maximum length = 128
tenant_id	<String>	Read-only	Tenant Id of the Config Jobs.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/sms_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sms_profile": {

<b>"to_list":<String_value></b>,
```

```
<b>"server_name":<String_value></b>,

<b>"profile_name":<String_value></b>,

"id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sms_profile":[{

"tenant_id":<String_value>,

"to_list":<String_value>,

"id":<String_value>,

"server_name":<String_value>,

"profile_name":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/sms_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/sms_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/sms_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of sms_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/sms_profile **? pagesize=#no;pageno=#no** Use this query-parameter to get the sms_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/sms_profile **?count=yes** Use this query-parameter to get the count of sms_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sms_profile":[{

"tenant_id":<String_value>,

"to_list":<String_value>,

"id":<String_value>,

"server_name":<String_value>,

"profile_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/sms_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sms_profile":[{

"tenant_id":<String_value>,

"to_list":<String_value>,
```

```
"id":<String_value>,

"server_name":<String_value>,

"profile_name":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/sms_profile/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sms_profile":{

<b>"id":<String_value></b>,

"to_list":<String_value>,

"server_name":<String_value>,

"profile_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sms_profile":[{

"tenant_id":<String_value>,

"to_list":<String_value>,

"id":<String_value>,

"server_name":<String_value>,

"profile_name":<String_value>}]}
```

sms_server

Configuration for SMS server properties resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
optional1_val	<String>	Read-write	Optional1 Val for the sms server. Maximum length = 128
to_seperater	<String>	Read-write	To list seperater for the sms server. Maximum length = 128
username_val	<String>	Read-write	Username val for the sms server. Maximum length = 128
to_key	<String>	Read-write	To key for the sms server.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
id	<String>	Read-write	Id is system generated key for all the sms server.
is_ssl	<Boolean>	Read-write	Is SSL support configured..
base_url	<String>	Read-write	Base URL for the sms server, without payload. Maximum length = 2000
username_key	<String>	Read-write	Username key for the sms server. Maximum length = 128
optional1_key	<String>	Read-write	Optional1 key for the sms server. Maximum length = 128
message_key	<String>	Read-write	Message key for the sms server. Maximum length = 128
message_word_sperater	<String>	Read-write	Message Word Seperater for the sms server. Maximum length = 128
type	<String>	Read-write	HTTP type supported for the sms server.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
password_val	<String>	Read-write	Password Val for the sms server. Maximum length = 128
password_key	<String>	Read-write	Password key for the sms server. Maximum length = 128
server_name	<String>	Read-write	SMS server name. Minimum length = 1 Maximum length = 128
tenant_id	<String>	Read-only	Tenant Id of the Config Jobs.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/sms_server?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sms_server": {

<b>"username_val":<String_value></b>,

<b>"to_key":<String_value></b>,

<b>"base_url":<String_value></b>,

<b>"username_key":<String_value></b>,

<b>"message_key":<String_value></b>,

<b>"type":<String_value></b>,

<b>"password_val":<String_value></b>,

<b>"password_key":<String_value></b>,

<b>"server_name":<String_value></b>,

"optional1_val":<String_value>,

"to_seperater":<String_value>,

"id":<String_value>,

"message_word_sperater":<String_value>,

"is_ssl":<Boolean_value>,

"optional1_key":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sms_server":[{
```

```
"optional2_val1":<String_value>,

"optional1_val":<String_value>,

"to_seperater":<String_value>,

"username_val":<String_value>,

"to_key":<String_value>,

"tenant_id":<String_value>,

"optional3_key":<String_value>,

"id":<String_value>,

"is_ssl":<Boolean_value>,

"optional3_val":<String_value>,

"base_url":<String_value>,

"username_key":<String_value>,

"optional2_key":<String_value>,

"optional1_key":<String_value>,

"message_key":<String_value>,

"message_word_sperater":<String_value>,

"type":<String_value>,

"password_val":<String_value>,

"password_key":<String_value>,

"server_name":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/sms_server/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/sms_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/sms_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of sms_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/sms_server **? pagesize=#no;pageno=#no** Use this query-parameter to get the sms_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/sms_server **?count=yes** Use this query-parameter to get the count of sms_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sms_server":[{

"optional2_val1":<String_value>,

"optional1_val":<String_value>,

"to_seperater":<String_value>,

"username_val":<String_value>,

"to_key":<String_value>,

"tenant_id":<String_value>,

"optional3_key":<String_value>,

"id":<String_value>,

"is_ssl":<Boolean_value>,

"optional3_val":<String_value>,

"base_url":<String_value>,

"username_key":<String_value>,

"optional2_key":<String_value>,

"optional1_key":<String_value>,
```

```
"message_key":<String_value>,

"message_word_sperater":<String_value>,

"type":<String_value>,

"password_val":<String_value>,

"password_key":<String_value>,

"server_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/sms_server/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sms_server":[{

"optional2_val1":<String_value>,

"optional1_val":<String_value>,
```

```
"to_seperater":<String_value>,

"username_val":<String_value>,

"to_key":<String_value>,

"tenant_id":<String_value>,

"optional3_key":<String_value>,

"id":<String_value>,

"is_ssl":<Boolean_value>,

"optional3_val":<String_value>,

"base_url":<String_value>,

"username_key":<String_value>,

"optional2_key":<String_value>,

"optional1_key":<String_value>,

"message_key":<String_value>,

"message_word_sperater":<String_value>,

"type":<String_value>,

"password_val":<String_value>,

"password_key":<String_value>,

"server_name":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/sms_server/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"sms_server":{

<b>"id":<String_value></b>,

"optional1_val":<String_value>,

"to_seperater":<String_value>,

"username_val":<String_value>,

"message_word_sperater":<String_value>,

"type":<String_value>,

"password_val":<String_value>,

"server_name":<String_value>,

"to_key":<String_value>,

"is_ssl":<Boolean_value>,

"base_url":<String_value>,

"username_key":<String_value>,

"message_key":<String_value>,

"optional1_key":<String_value>,

"password_key":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"sms_server":[{

"optional2_val1":<String_value>,

"optional1_val":<String_value>,

"to_seperater":<String_value>,

"username_val":<String_value>,

"to_key":<String_value>,

"tenant_id":<String_value>,

"optional3_key":<String_value>,

"id":<String_value>,

"is_ssl":<Boolean_value>,

"optional3_val":<String_value>,

"base_url":<String_value>,

"username_key":<String_value>,

"optional2_key":<String_value>,

"optional1_key":<String_value>,

"message_key":<String_value>,

"message_word_sperater":<String_value>,

"type":<String_value>,
```

```
"password_val":<String_value>,

"password_key":<String_value>,

"server_name":<String_value>}]}
```

smtp_server

Configuration for SMTP server properties resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
sender_mail_id	<String>	Read-write	Email Address from which email is to be sent.
is_auth	<Boolean>	Read-write	Is authentication enabled for this smtp server.
port	<Integer>	Read-write	SMTP Server port address.. Minimum value = 1 Maximum value =
username	<String>	Read-write	Username for the smtp server. Maximum length = 128
password	<String>	Read-write	Password for the smtp server. Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the smtp server.
is_ssl	<Boolean>	Read-write	Is this smtp server is SSL support configured..
server_name	<String>	Read-write	SMTP server name. Minimum length = 1 Maximum length = 128
tenant_id	<String>	Read-only	Tenant Id of the Config Jobs.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/smtp_server?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smtp_server": {

<b>"server_name":<String_value></b>,
```

```
"sender_mail_id":<String_value>,

"password":<String_value>,

"id":<String_value>,

"port":<Integer_value>,

"username":<String_value>,

"is_ssl":<Boolean_value>,

"is_auth":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smtp_server":[{

"sender_mail_id":<String_value>,

"is_auth":<Boolean_value>,

"port":<Integer_value>,

"username":<String_value>,

"password":<String_value>,

"tenant_id":<String_value>,

"id":<String_value>,

"is_ssl":<Boolean_value>,

"server_name":<String_value>}]}
```

```

```

delete

URL: https://<MGMT-IP>/nitro/v2/config/smtp_server/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/smtp_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/smtp_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of smtp_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/smtp_server **? pagesize=#no;pageno=#no** Use this query-parameter to get the smtp_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/smtp_server **?count=yes** Use this query-parameter to get the count of smtp_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smtp_server":[{

"sender_mail_id":<String_value>,

"is_auth":<Boolean_value>,

"port":<Integer_value>,

"username":<String_value>,

"password":<String_value>,

"tenant_id":<String_value>,

"id":<String_value>,

"is_ssl":<Boolean_value>,

"server_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/smtp_server/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smtp_server":[{

"sender_mail_id":<String_value>,

"is_auth":<Boolean_value>,

"port":<Integer_value>,

"username":<String_value>,

"password":<String_value>,

"tenant_id":<String_value>,

"id":<String_value>,

"is_ssl":<Boolean_value>,

"server_name":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/smtp_server/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"smtp_server":{

<b>"id":<String_value></b>,

"sender_mail_id":<String_value>,

"password":<String_value>,

"port":<Integer_value>,

"username":<String_value>,

"server_name":<String_value>,

"is_ssl":<Boolean_value>,

"is_auth":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"smtp_server":[{

"sender_mail_id":<String_value>,

"is_auth":<Boolean_value>,
```

```
"port":<Integer_value>,

"username":<String_value>,

"password":<String_value>,

"tenant_id":<String_value>,

"id":<String_value>,

"is_ssl":<Boolean_value>,

"server_name":<String_value>}]}
```

snmp_alarm_config

Configuration for SNMP Alarm Configurations resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
enable	<Boolean>	Read-write	Enable Alarm.
threshold	<Integer>	Read-write	Threshold Value for the alarm. Maximum value =
time	<Integer>	Read-write	Frequency of the alarm in minutes.
name	<String>	Read-write	Alarm Name. Maximum length = 128
severity	<String>	Read-write	Alarm severity. Supported values: Critical, Major, Minor,

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Warning, Informational . Maximum length = 128
clear_alarm_oid	<Integer>	Read-only	Clear alarm Oid.

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/snmp_alarm_config **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/snmp_alarm_config **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of snmp_alarm_config resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/snmp_alarm_config **? pagesize=#no;pageno=#no** Use this query-parameter to get the snmp_alarm_config resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/snmp_alarm_config **?count=yes** Use this query-parameter to get the count of snmp_alarm_config resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_alarm_config":[{

"alarm_id":<Integer_value>,

"enable":<Boolean_value>,

"threshold":<Integer_value>,

"clear_alarm_oid":<Integer_value>,

"time":<Integer_value>,

"name":<String_value>,

"severity":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/snmp_alarm_config/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_alarm_config":[{

"alarm_id":<Integer_value>,

"enable":<Boolean_value>,

"threshold":<Integer_value>,

"clear_alarm_oid":<Integer_value>,

"time":<Integer_value>,

"name":<String_value>,

"severity":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/snmp_alarm_config/name_value<String> **HTTPS**
Method: PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"snmp_alarm_config":{

"name":<String_value>,

"threshold":<Integer_value>,

"time":<Integer_value>,

"severity":<String_value>,
```

```
"enable":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_alarm_config":[{

"alarm_id":<Integer_value>,

"enable":<Boolean_value>,

"threshold":<Integer_value>,

"clear_alarm_oid":<Integer_value>,

"time":<Integer_value>,

"name":<String_value>,

"severity":<String_value>}]}
```

snmp_info

Configuration for SNMP Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
engine_id	<String>	Read-write	SNMP EngineID. Maximum length = 64

Operations

(click to see Properties)

- GET (ALL)
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/snmp_info **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/snmp_info **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of snmp_info resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/snmp_info **? pagesize=#no;pageno=#no** Use this query-parameter to get the snmp_info resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/snmp_info **?count=yes** Use this query-parameter to get the count of snmp_info resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_info":[{

"engine_id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/snmp_info/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"snmp_info":{

<b>"engine_id":<String_value></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_info":[{

"engine_id":<String_value>}]}
```

snmp_manager

Configuration for SNMP Agent Manager configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
community	<String>	Read-write	Community Name. Minimum length = 1 Maximum length = 128
netmask	<String>	Read-write	Netmask. Minimum length = 1 Maximum length = 64
ip_address	<String>	Read-write	Manager IPAddress.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/snmp_manager?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"snmp_manager": {

<b>"community":<String_value></b>,

<b>"ip_address":<String_value></b>,

"netmask":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_manager":[{

"community":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"address_type":<Integer_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/snmp_manager/ip_address_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/snmp_manager **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/snmp_manager **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of snmp_manager resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/snmp_manager **? pagesize=#no;pageno=#no** Use this query-parameter to get the snmp_manager resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/snmp_manager **?count=yes** Use this query-parameter to get the count of snmp_manager resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_manager":[{

"community":<String_value>,

"netmask":<String_value>,
```

```
"ip_address":<String_value>,

"address_type":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/snmp_manager/ip_address_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_manager":[{

"community":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"address_type":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/snmp_manager/ip_address_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"snmp_manager":{

<b>"community":<String_value></b>,

<b>"ip_address":<String_value></b>,

"netmask":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_manager":[{

"community":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"address_type":<Integer_value>}]}
```

snmp_mib

Configuration for SNMP MIB Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
custom_id	<String>	Read-write	Custom identification number for appliance. Minimum length = 1 Maximum length = 127
id	<String>	Read-write	Id is system generated key.
location	<String>	Read-write	Physical location of appliance. Minimum length = 1 Maximum length = 127
name	<String>	Read-write	Name for appliance. Minimum length = 1 Maximum length = 127
contact	<String>	Read-write	Name of the administrator for appliance.. Minimum length = 1 Maximum length = 127

Operations

(click to see Properties)

- GET (ALL)
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/snmp_mib **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/snmp_mib **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of snmp_mib resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/snmp_mib **?pagesize=#no;pageno=#no** Use this query-parameter to get the snmp_mib resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/snmp_mib **?count=yes** Use this query-parameter to get the count of snmp_mib resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_mib":[{

"custom_id":<String_value>,

"id":<String_value>,

"location":<String_value>,
```

```
"name":<String_value>,

"contact":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/snmp_mib/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"snmp_mib":{

<b>"location":<String_value></b>,

<b>"name":<String_value></b>,

<b>"contact":<String_value></b>,

"id":<String_value>,

"custom_id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_mib":[{

"custom_id":<String_value>,

"id":<String_value>,

"location":<String_value>,

"name":<String_value>,

"contact":<String_value>}]}
```

snmp_trap

Configuration for SNMP Trap Destinations resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
community	<String>	Read-write	Community Name. Maximum length = 32
version	<String>	Read-write	SNMP version. Maximum length = 2
dest_server	<String>	Read-write	Trap Destination Server Address.
dest_port	<Integer>	Read-write	Destination Port. Minimum value = 1 Maximum value =
user_name	<String[]>	Read-write	Name of SNMP Trap User.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 32

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/snmp_trap?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"snmp_trap": {

<b>"dest_server":<String_value></b>,

"version":<String_value>,

"user_name":<String_value[]>,

"community":<String_value>,

"dest_port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_trap":[{

"community":<String_value>,

"version":<String_value>,

"dest_server":<String_value>,

"dest_port":<Integer_value>,

"user_name":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/snmp_trap/dest_server_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/snmp_trap **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/snmp_trap **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of

snmp_trap resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/snmp_trap **?pagesize=#no;pageno=#no** Use this query-parameter to get the snmp_trap resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/snmp_trap **?count=yes** Use this query-parameter to get the count of snmp_trap resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_trap":[{

"community":<String_value>,

"version":<String_value>,

"dest_server":<String_value>,

"dest_port":<Integer_value>,

"user_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/snmp_trap/dest_server_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_trap":[{

"community":<String_value>,

"version":<String_value>,

"dest_server":<String_value>,

"dest_port":<Integer_value>,

"user_name":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/snmp_trap/dest_server_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"snmp_trap":{

<b>"dest_server":<String_value></b>,

"version":<String_value>,

"user_name":<String_value[]>,

"community":<String_value>,

"dest_port":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_trap":[{

"community":<String_value>,

"version":<String_value>,

"dest_server":<String_value>,
```

```
"dest_port":<Integer_value>,

"user_name":<String_value>}]}
```

snmp_user

Configuration for SNMP User resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
security_level	<Integer>	Read-write	Security Level of SNMP User. Values: 0: noAuthNoPriv, 1: authNoPriv, 2: authPriv. Maximum value =
auth_password	<String>	Read-write	Authentication Password of SNMP User. Minimum length = 8 Maximum length = 32
privacy_protocol	<Integer>	Read-write	Privacy Protocol of SNMP User. Values: 0:noValue, 1: DES, 2: AES. Maximum value =
name	<String>	Read-write	Name of SNMP User. Minimum length = 1 Maximum length = 32

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
privacy_password	<String>	Read-write	Privacy Password of SNMP User. Minimum length = 8 Maximum length = 32
view_name	<String>	Read-write	SNMP View Name attached to the SNMP User. Maximum length = 32
auth_protocol	<Integer>	Read-write	Authentication Protocol of SNMP User. Values: 0:noValue, 1: MD5, 2: SHA1. Maximum value =

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/snmp_user?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"snmp_user": {

<b>"security_level":<Integer_value></b>,

<b>"name":<String_value></b>,

"privacy_password":<String_value>,

"auth_protocol":<Integer_value>,

"view_name":<String_value>,

"auth_password":<String_value>,

"privacy_protocol":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_user":[{

"security_level":<Integer_value>,

"auth_password":<String_value>,

"privacy_protocol":<Integer_value>,

"name":<String_value>,

"privacy_password":<String_value>,

"view_name":<String_value>,

"auth_protocol":<Integer_value>,

"security_level_str":<String_value>,
```

```
"privacy_protocol_str":<String_value>,

"auth_protocol_str":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/snmp_user/name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/snmp_user **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/snmp_user **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of snmp_user resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/snmp_user **? pagesize=#no;pageno=#no** Use this query-parameter to get the snmp_user resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/snmp_user **?count=yes** Use this query-parameter to get the count of snmp_user resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_user":[{

"security_level":<Integer_value>,

"auth_password":<String_value>,

"privacy_protocol":<Integer_value>,

"name":<String_value>,

"privacy_password":<String_value>,

"view_name":<String_value>,

"auth_protocol":<Integer_value>,

"security_level_str":<String_value>,

"privacy_protocol_str":<String_value>,

"auth_protocol_str":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/snmp_user/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_user":[{

"security_level":<Integer_value>,

"auth_password":<String_value>,

"privacy_protocol":<Integer_value>,

"name":<String_value>,

"privacy_password":<String_value>,

"view_name":<String_value>,

"auth_protocol":<Integer_value>,

"security_level_str":<String_value>,

"privacy_protocol_str":<String_value>,
```

```
"auth_protocol_str":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/snmp_user/name_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"snmp_user":{

<b>"security_level":<Integer_value></b>,

<b>"name":<String_value></b>,

"privacy_password":<String_value>,

"auth_protocol":<Integer_value>,

"view_name":<String_value>,

"auth_password":<String_value>,

"privacy_protocol":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_user":[{

"security_level":<Integer_value>,

"auth_password":<String_value>,

"privacy_protocol":<Integer_value>,

"name":<String_value>,

"privacy_password":<String_value>,

"view_name":<String_value>,

"auth_protocol":<Integer_value>,

"security_level_str":<String_value>,

"privacy_protocol_str":<String_value>,

"auth_protocol_str":<String_value>}]}
```

snmp_view

Configuration for SNMP view resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
subtree	<String>	Read-write	Subtree associated with the SNMP view.A particular branch (subtree) of the MIB tree that you want to associate with this view.You must specify the

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			subtree as an SNMP OID. Minimum length = 1 Maximum length = 256
name	<String>	Read-write	Name of SNMP view. Minimum length = 1 Maximum length = 32
type	<Boolean>	Read-write	Include or Exclude the associated subtree . Values. true:Include, false: Exclude.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/snmp_view?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"snmp_view": {

<b>"name":<String_value></b>,
```

```
"subtree":<String_value>,

"type":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_view":[{

"subtree":<String_value>,

"name":<String_value>,

"type":<Boolean_value>,

"type_str":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/snmp_view/name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/snmp_view **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/snmp_view **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of snmp_view resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/snmp_view **? pagesize=#no;pageno=#no** Use this query-parameter to get the snmp_view resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/snmp_view **?count=yes** Use this query-parameter to get the count of snmp_view resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_view":[{

"subtree":<String_value>,
```

```
"name":<String_value>,

"type":<Boolean_value>,

"type_str":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/snmp_view/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_view":[{

"subtree":<String_value>,

"name":<String_value>,

"type":<Boolean_value>,

"type_str":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/snmp_view/name_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"snmp_view":{

<b>"name":<String_value></b>,

"subtree":<String_value>,

"type":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"snmp_view":[{

"subtree":<String_value>,

"name":<String_value>,

"type":<Boolean_value>,

"type_str":<String_value>}]}
```

ssl_cert

Configuration for SSL certificate File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/ssl_cert/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ssl_cert **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ssl_cert **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ssl_cert resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ssl_cert **? pagesize=#no;pageno=#no** Use this query-parameter to get the ssl_cert resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ssl_cert **?count=yes** Use this query-parameter to get the count of ssl_cert resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ssl_cert":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_dir":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ssl_cert/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ssl_cert":[{

"file_location_path":<String_value>,
```

```
"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_dir":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/ssl_cert **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

ssl_key

Configuration for SSL key File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/ssl_key/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ssl_key **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ssl_key **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ssl_key resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ssl_key **? pagesize=#no;pageno=#no** Use this query-parameter to get the ssl_key resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ssl_key **?count=yes** Use this query-parameter to get the count of ssl_key resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ssl_key":[{
```

```
"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_dir":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/ssl_key/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ssl_key":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,
```

```
"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_dir":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/ssl_key **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

ssl_settings

Configuration for SSL Settings resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
sslreneg	<Boolean>	Read-write	Enable SSL Renegotiation.
tlsv1	<Boolean>	Read-write	Enable TLSv1.
sslv3	<Boolean>	Read-write	Enable SSLv3.
tlsv1_1	<Boolean>	Read-write	Enable TLSv1.1.
tlsv1_2	<Boolean>	Read-write	Enable TLSv1.2.

Operations

(click to see Properties)

- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get

URL: https://<MGMT-IP>/nitro/v2/config/ssl_settings **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ssl_settings":[{

"sslreneg":<Boolean_value>,

"tlsv1":<Boolean_value>,
```

```
"sslv3":<Boolean_value>,

"tlsv1_1":<Boolean_value>,

"tlsv1_2":<Boolean_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/ssl_settings/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"ssl_settings":{

"sslreneg":<Boolean_value>,

"tlsv1_2":<Boolean_value>,

"sslv3":<Boolean_value>,

"tlsv1":<Boolean_value>,

"tlsv1_1":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ssl_settings":[{

"sslreneg":<Boolean_value>,

"tlsv1":<Boolean_value>,

"sslv3":<Boolean_value>,

"tlsv1_1":<Boolean_value>,

"tlsv1_2":<Boolean_value>}]}
```

static_route

Configuration for Static Route resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
network	<String>	Read-write	network for route addition.
gateway	<String>	Read-write	Gateway for route addition.
netmask	<String>	Read-write	netmask for route addition.
id	<String>	Read-write	Id is system generated key.

Operations

(click to see Properties)

- ADD
- DELETE

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/static_route?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"static_route": {

"network":<String_value>,

"id":<String_value>,

"gateway":<String_value>,

"netmask":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"static_route":[{

"network":<String_value>,

"gateway":<String_value>,

"netmask":<String_value>,

"id":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/static_route/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/static_route **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/static_route **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of static_route resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/static_route **? pagesize=#no;pageno=#no** Use this query-parameter to get the static_route resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/static_route **?count=yes** Use this query-parameter to get the count of static_route resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"static_route":[{

"network":<String_value>,

"gateway":<String_value>,

"netmask":<String_value>,

"id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/static_route/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"static_route":[{

"network":<String_value>,

"gateway":<String_value>,

"netmask":<String_value>,

"id":<String_value>}]}
```

syslog_facility

Configuration for Syslog filter facility details resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-only	facility name.
code	<String>	Read-only	facility code.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/syslog_facility **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/syslog_facility **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of syslog_facility resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/syslog_facility **? pagesize=#no;pageno=#no** Use this query-parameter to get the syslog_facility resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/syslog_facility **?count=yes** Use this query-parameter to get the count of syslog_facility resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_facility":[{

"name":<String_value>,

"code":<String_value>}]}
```

syslog_messages

Configuration for System Log Message resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
source	<String>	Read-write	Source IP Address.
event_type	<String>	Read-write	Event type.
message	<String>	Read-write	Message.
app_name	<String>	Read-write	Application Name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
system_time	<String>	Read-write	System Time in local timezone.
id	<String>	Read-write	Id is system generated key for all the system log entries.
instance_ip	<String>	Read-write	Instance IP Address.
severity	<String>	Read-write	Severity of the message.
module	<String>	Read-write	Module name.
system_gmt_time	<Integer>	Read-only	System Time in GMT timezone.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/syslog_messages **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/syslog_messages **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of syslog_messages resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/syslog_messages **?pagesize=#no;pageno=#no** Use this query-parameter to get the syslog_messages resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/syslog_messages **?count=yes** Use this query-parameter to get the count of syslog_messages resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_messages":[{

"source":<String_value>,

"event_type":<String_value>,

"system_gmt_time":<Integer_value>,

"message":<String_value>,

"app_name":<String_value>,

"system_time":<String_value>,

"id":<String_value>,

"instance_ip":<String_value>,

"severity":<String_value>,

"module":<String_value>}]}
```

syslog_params

Configuration for Syslog Parameters resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
date_format	<String>	Read-write	Format of date to be added in the syslog message.
timezone	<String>	Read-write	Timezone to be used in the syslog message.
id	<String>	Read-write	Id is system generated key.

Operations

(click to see Properties)

- GET (ALL)
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/syslog_params **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/syslog_params **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of syslog_params resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/syslog_params **?pagesize=#no;pageno=#no** Use this query-parameter to get the syslog_params resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/syslog_params **?count=yes** Use this query-parameter to get the count of syslog_params resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_params":[{

"date_format":<String_value>,

"timezone":<String_value>,

"id":<String_value>}]}
```

modify

URL: https://<MGMT-IP>/nitro/v2/config/syslog_params/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"syslog_params":{

"date_format":<String_value>,

"timezone":<String_value>,

"id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_params":[{

"date_format":<String_value>,

"timezone":<String_value>,

"id":<String_value>}]}
```

syslog_persistant_stat_info

Configuration for This table holds stat info to be persistant across reboots for event feature. resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
propkey	<String>	Read-write	Key of the property.. Minimum length = 1 Maximum length = 256
id	<String>	Read-write	Id is system generated key for all the event stat info entries.
propvalue	<String>	Read-write	Value of the property.. Maximum length = 2000

Operations

(click to see Properties)

- DELETE
- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/syslog_persistant_stat_info/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/syslog_persistant_stat_info **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/syslog_persistant_stat_info **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of syslog_persistant_stat_info resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/syslog_persistant_stat_info **? pagesize=#no;pageno=#no** Use this query-parameter to get the syslog_persistant_stat_info resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/syslog_persistant_stat_info **?count=yes** Use this query-parameter to get the count of syslog_persistant_stat_info resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_persistant_stat_info":[{

"propkey":<String_value>,

"id":<String_value>,

"propvalue":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/syslog_persistant_stat_info/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_persistant_stat_info":[{

"propkey":<String_value>,

"id":<String_value>,

"propvalue":<String_value>}]}
```

syslog_purge_setting

Configuration for Syslog Purge Setting resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
purge_interval_for_appfw_data	<Integer>	Read-write	purge_interval_for_appfw_data.
id	<String>	Read-write	Id is system generated key for all the events.
details	<syslog_purge_setting_details[]>	Read-write	Syslog Purge Setting Details.
purge_interval_for_generic_data	<Integer>	Read-write	purge_interval_for_generic_data.
purge_interval_for_gateway_data	<Integer>	Read-write	purge_interval_for_gateway_data..

Operations

(click to see Properties)

- GET (ALL)
- GET

- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/syslog_purge_setting **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/syslog_purge_setting **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of

syslog_purge_setting resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/syslog_purge_setting **?pagesize=#no;pageno=#no** Use this query-parameter to get the syslog_purge_setting resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/syslog_purge_setting **?count=yes** Use this query-parameter to get the count of syslog_purge_setting resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_purge_setting":[{

"purge_interval_for_appfw_data":<Integer_value>,

"id":<String_value>,

"details":[{

"syslog_detail_type_desc":<String_value>,

"parent_name":<String_value>,
```

```
"syslog_type":<String_value>,

"syslog_module":<String_value>,

"id":<String_value>,

"syslog_detail_purge_interval":<Integer_value>,

"parent_id":<String_value>}],

"purge_interval_for_generic_data":<Integer_value>,

"purge_interval_for_gateway_data":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/syslog_purge_setting/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_purge_setting":[{
```

```
"purge_interval_for_appfw_data":<Integer_value>,

"id":<String_value>,

"details":[{

"syslog_detail_type_desc":<String_value>,

"parent_name":<String_value>,

"syslog_type":<String_value>,

"syslog_module":<String_value>,

"id":<String_value>,

"syslog_detail_purge_interval":<Integer_value>,

"parent_id":<String_value>}],

"purge_interval_for_generic_data":<Integer_value>,

"purge_interval_for_gateway_data":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/syslog_purge_setting/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"syslog_purge_setting":{

<b>"id":<String_value></b>,

"purge_interval_for_appfw_data":<Integer_value>,

"details":[{

<b>"id":<String_value></b>,

"parent_id":<String_value>,

"syslog_detail_type_desc":<String_value>,

"syslog_module":<String_value>,

"parent_name":<String_value>,

"syslog_detail_purge_interval":<Integer_value>,

"syslog_type":<String_value>}],

"purge_interval_for_gateway_data":<Integer_value>,

"purge_interval_for_generic_data":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_purge_setting":[{

"purge_interval_for_appfw_data":<Integer_value>,

"id":<String_value>,

"details":[{
```

```
"syslog_detail_type_desc":<String_value>,

"parent_name":<String_value>,

"syslog_type":<String_value>,

"syslog_module":<String_value>,

"id":<String_value>,

"syslog_detail_purge_interval":<Integer_value>,

"parent_id":<String_value>}],

"purge_interval_for_generic_data":<Integer_value>,

"purge_interval_for_gateway_data":<Integer_value>}]}
```

syslog_purge_setting_details

Configuration for Syslog Purge Setting Details resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
syslog_detail_type_desc	<String>	Read-write	syslog groups. Maximum length = 255
parent_name	<String>	Read-write	.
syslog_type	<String>	Read-write	syslog groups. Maximum length = 255
syslog_module	<String>	Read-write	syslog modulename.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 255
id	<String>	Read-write	Id is system generated key for all the events.
syslog_detail_purge_interval	<Integer>	Read-write	Purge interval for syslog groups..
parent_id	<String>	Read-write	ID of syslog_purge_setting_details.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

syslog_server

Configuration for Syslog Server resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
log_level_all	<Boolean>	Read-write	Send logs of all levels to this syslog server.
name	<String>	Read-write	Syslog server name. Minimum length = 1 Maximum length = 128
log_level_warning	<Boolean>	Read-write	Send logs of level warning to this syslog server.
log_level_error	<Boolean>	Read-write	Send logs of level error to this syslog server.
port	<Integer>	Read-write	Syslog server port. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
log_level_info	<Boolean>	Read-write	Send logs of level info to this syslog server.
log_level_critical	<Boolean>	Read-write	Send logs of level critical to this syslog server.
ip_address	<String>	Read-write	Syslog server IP address. Minimum length = 1 Maximum length = 64
log_level_none	<Boolean>	Read-write	Send no logs to this syslog server.
log_levels	<String>	Read-only	Set of all log levels at which messages are sent to this syslog server.
address_type	<Integer>	Read-only	Configuration Type. Values: 0: IPv4, 1: IPv6.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- MODIFY

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/syslog_server?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"syslog_server": {

<b>"name":<String_value></b>,

<b>"port":<Integer_value></b>,

<b>"ip_address":<String_value></b>,

"log_level_all":<Boolean_value>,

"log_level_info":<Boolean_value>,

"log_level_critical":<Boolean_value>,

"log_level_none":<Boolean_value>,

"log_level_error":<Boolean_value>,

"log_level_warning":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_server":[{

"log_level_all":<Boolean_value>,

"log_level_alert":<Boolean_value>,

"name":<String_value>,

"log_level_debug":<Boolean_value>,
```

```
"log_level_warning":<Boolean_value>,

"log_level_error":<Boolean_value>,

"port":<Integer_value>,

"log_levels":<String_value>,

"log_level_info":<Boolean_value>,

"log_level_notice":<Boolean_value>,

"log_level_critical":<Boolean_value>,

"ip_address":<String_value>,

"log_level_none":<Boolean_value>,

"address_type":<Integer_value>,

"log_level_emergency":<Boolean_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/syslog_server/name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/syslog_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/syslog_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of syslog_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/syslog_server **? pagesize=#no;pageno=#no** Use this query-parameter to get the syslog_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/syslog_server **?count=yes** Use this query-parameter to get the count of syslog_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_server":[{

"log_level_all":<Boolean_value>,

"log_level_alert":<Boolean_value>,

"name":<String_value>,

"log_level_debug":<Boolean_value>,

"log_level_warning":<Boolean_value>,

"log_level_error":<Boolean_value>,

"port":<Integer_value>,

"log_levels":<String_value>,

"log_level_info":<Boolean_value>,

"log_level_notice":<Boolean_value>,

"log_level_critical":<Boolean_value>,

"ip_address":<String_value>,

"log_level_none":<Boolean_value>,

"address_type":<Integer_value>,

"log_level_emergency":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/syslog_server/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_server":[{

"log_level_all":<Boolean_value>,

"log_level_alert":<Boolean_value>,

"name":<String_value>,

"log_level_debug":<Boolean_value>,

"log_level_warning":<Boolean_value>,

"log_level_error":<Boolean_value>,

"port":<Integer_value>,

"log_levels":<String_value>,

"log_level_info":<Boolean_value>,

"log_level_notice":<Boolean_value>,

"log_level_critical":<Boolean_value>,

"ip_address":<String_value>,

"log_level_none":<Boolean_value>,

"address_type":<Integer_value>,
```

```
"log_level_emergency":<Boolean_value>}]}
```

modify

URL: https://<MGMT-IP>/nitro/v2/config/syslog_server/name_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"syslog_server":{

"log_level_all":<Boolean_value>,

"name":<String_value>,

"log_level_error":<Boolean_value>,

"log_level_warning":<Boolean_value>,

"port":<Integer_value>,

"log_level_info":<Boolean_value>,

"log_level_critical":<Boolean_value>,

"log_level_none":<Boolean_value>,

"ip_address":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"syslog_server":[{

"log_level_all":<Boolean_value>,

"log_level_alert":<Boolean_value>,

"name":<String_value>,

"log_level_debug":<Boolean_value>,

"log_level_warning":<Boolean_value>,

"log_level_error":<Boolean_value>,

"port":<Integer_value>,

"log_levels":<String_value>,

"log_level_info":<Boolean_value>,

"log_level_notice":<Boolean_value>,

"log_level_critical":<Boolean_value>,

"ip_address":<String_value>,

"log_level_none":<Boolean_value>,

"address_type":<Integer_value>,

"log_level_emergency":<Boolean_value>}]}
```

system_settings

Configuration for System Settings resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
svm_ns_comm	<String>	Read-write	Communication with Instances. Minimum length = 1 Maximum length = 10
session_timeout	<Integer>	Read-write	Session timeout for the system.
session_timeout_unit	<String>	Read-write	Session timeout unit for the system.
enable_nsrecover_login	<Boolean>	Read-write	This setting enalbes nsrecover login for SVM.
enable_session_timeout	<Boolean>	Read-write	Enables session timeout feature.
id	<String>	Read-write	Id is system generated key.
enable_certificate_download	<Boolean>	Read-write	Enable Certificate Download.
keep_alive_ping_interval	<Integer>	Read-write	Agent web socket keep alive ping interval for the system.
enable_apiproxy_credentials	<Boolean>	Read-write	Enable API Proxy Credentials.
enable_shell_access	<Boolean>	Read-write	Enable Shell access for non-nsroot User(s).
basicauth	<Boolean>	Read-write	Allow Basic Authentication Protocol.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
secure_access_only	<Boolean>	Read-write	Secure Access only.
is_metering_enabled	<Boolean>	Read-write	Enable Metering for Citrix ADC VPX on SDX.
enable_delete_interface_on_adc	<Boolean>	Read-write	Flag to enable/ disable deleting interface from ADCs on SDX.
port_model	<Integer>	Read-only	Port Model for BR instance.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get

URL: https://<MGMT-IP>/nitro/v2/config/system_settings **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"system_settings":[{

"svm_ns_comm":<String_value>,
```

```
"session_timeout":<Integer_value>,

"session_timeout_unit":<String_value>,

"enable_nsrecover_login":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"port_model":<Integer_value>,

"id":<String_value>,

"ns_br_interface":<String_value>,

"enable_certificate_download":<Boolean_value>,

"keep_alive_ping_interval":<Integer_value>,

"init_status":<Integer_value>,

"enable_apiproxy_credentials":<Boolean_value>,

"ns_br_interface_2":<String_value>,

"enable_shell_access":<Boolean_value>,

"basicauth":<Boolean_value>,

"secure_access_only":<Boolean_value>,

"vm_auto_poweron":<Boolean_value>,

"is_metering_enabled":<Boolean_value>,

"enable_delete_interface_on_adc":<Boolean_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>,

"reset":<Boolean_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/system_settings/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"system_settings":{

"session_timeout":<Integer_value>,

"session_timeout_unit":<String_value>,

"enable_session_timeout":<Boolean_value>,

"id":<String_value>,

"keep_alive_ping_interval":<Integer_value>,

"enable_shell_access":<Boolean_value>,

"basicauth":<Boolean_value>,

"secure_access_only":<Boolean_value>,

"svm_ns_comm":<String_value>,

"enable_nsrecover_login":<Boolean_value>,

"enable_certificate_download":<Boolean_value>,

"enable_apiproxy_credentials":<Boolean_value>,
```

```
"is_metering_enabled":<Boolean_value>,

"enable_delete_interface_on_adc":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"system_settings":[{

"svm_ns_comm":<String_value>,

"session_timeout":<Integer_value>,

"session_timeout_unit":<String_value>,

"enable_nsrecover_login":<Boolean_value>,

"enable_session_timeout":<Boolean_value>,

"port_model":<Integer_value>,

"id":<String_value>,

"ns_br_interface":<String_value>,

"enable_certificate_download":<Boolean_value>,

"keep_alive_ping_interval":<Integer_value>,

"init_status":<Integer_value>,

"enable_apiproxy_credentials":<Boolean_value>,

"ns_br_interface_2":<String_value>,

"enable_shell_access":<Boolean_value>,
```

```
"basicauth":<Boolean_value>,

"secure_access_only":<Boolean_value>,

"vm_auto_poweron":<Boolean_value>,

"is_metering_enabled":<Boolean_value>,

"enable_delete_interface_on_adc":<Boolean_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>,

"reset":<Boolean_value>}]}
```

system_threshold

Configuration for System Thresholds resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	name.
region	<String>	Read-write	Region Code. Maximum length = 256
rule	<String>	Read-write	Rule.
stylebook_flag	<Boolean>	Read-write	Stylebook threshold configuration true or false.
country	<String>	Read-write	Country Code. Maximum length = 256

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
sms_profile	<String>	Read-write	SMS Profile.
slack_profile	<String>	Read-write	Slack Profile.
feature	<String>	Read-write	feature.
ctnsappname	<String>	Read-write	AppName.
id	<String>	Read-write	Id is system generated key for all the Thresholds configuration . Maximum length = 256
resource_type	<String>	Read-write	Resource Type.
mail_profile	<String>	Read-write	Mail Profile.
is_enabled	<String>	Read-write	true or false.
pagerduty_profile	<String>	Read-write	PagerDuty Profile.
duration	<String>	Read-write	duration of metric to be checked against threshold.
servicenow_profile	<String>	Read-write	Servicenow Profile.
city	<String>	Read-write	City. Maximum length = 256
ip_address	<String>	Read-write	Field to store the ip address as it is along with any extension. Maximum length = 512
group_name	<String>	Read-write	Group Name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
reference_key	<String>	Read-write	Reference Key.
exporter_id	<Double>	Read-write	Exporter Id.
category_list	<String[]>	Read-write	List of selected event category..
tenant_name	<String>	Read-only	Tenant Name.

Operations

(click to see Properties)

- GET (ALL)
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/system_threshold **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/system_threshold **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of system_threshold resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/system_threshold **?pagesize=#no;pageno=#no** Use this query-parameter to get the system_threshold resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/system_threshold **?count=yes** Use this query-parameter to get the count of system_threshold resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"system_threshold":[{

"name":<String_value>,

"region":<String_value>,

"rule":<String_value>,

"stylebook_flag":<Boolean_value>,

"country":<String_value>,

"sms_profile":<String_value>,

"slack_profile":<String_value>,

"feature":<String_value>,

"ctnsappname":<String_value>,

"id":<String_value>,

"resource_type":<String_value>,

"mail_profile":<String_value>,

"is_enabled":<String_value>,

"pagerduty_profile":<String_value>,

"duration":<String_value>,

"servicenow_profile":<String_value>,

"city":<String_value>,

"ip_address":<String_value>,

"tenant_name":<String_value>,
```

```
"group_name":<String_value>,

"reference_key":<String_value>,

"exporter_id":<Double_value>,

"category_list":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/system_threshold/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"system_threshold":{

"sms_profile":<String_value>,

"slack_profile":<String_value>,

"feature":<String_value>,

"ctnsappname":<String_value>,

"id":<String_value>,

"mail_profile":<String_value>,

"resource_type":<String_value>,

"is_enabled":<String_value>,
```

```
"pagerduty_profile":<String_value>,

"region":<String_value>,

"name":<String_value>,

"duration":<String_value>,

"servicenow_profile":<String_value>,

"stylebook_flag":<Boolean_value>,

"rule":<String_value>,

"city":<String_value>,

"ip_address":<String_value>,

"country":<String_value>,

"group_name":<String_value>,

"reference_key":<String_value>,

"category_list":<String_value[]>,

"exporter_id":<Double_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"system_threshold":[{

"name":<String_value>,

"region":<String_value>,
```

```
"rule":<String_value>,

"stylebook_flag":<Boolean_value>,

"country":<String_value>,

"sms_profile":<String_value>,

"slack_profile":<String_value>,

"feature":<String_value>,

"ctnsappname":<String_value>,

"id":<String_value>,

"resource_type":<String_value>,

"mail_profile":<String_value>,

"is_enabled":<String_value>,

"pagerduty_profile":<String_value>,

"duration":<String_value>,

"servicenow_profile":<String_value>,

"city":<String_value>,

"ip_address":<String_value>,

"tenant_name":<String_value>,

"group_name":<String_value>,

"reference_key":<String_value>,

"exporter_id":<Double_value>,

"category_list":<String_value>}]}
```

tacacs_server

Configuration for TACACS Server configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
tacacs_key	<String>	Read-write	Key shared between the TACACS+ server and clients. Minimum length = 1 Maximum length = 64
auth_timeout	<Integer>	Read-write	The maximum number of seconds the system will wait for a response from the TACACS server. Minimum value = 1 Maximum value =
name	<String>	Read-write	Name of TACACS server. Minimum length = 1 Maximum length = 128
group_attr_name	<String>	Read-write	The Attribute name for group extraction from the ACS server. If not passed, then groups will not be extracted. No other harm.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
port	<Integer>	Read-write	port number of TACACS server. Minimum value = 1 Maximum value =
id	<String>	Read-write	Id is system generated key for all the TACACS servers.
ip_address	<String>	Read-write	IP Address of TACACS server. Minimum length = 1 Maximum length = 64
accounting	<Boolean>	Read-write	Enable accounting in the tacacs server.
address_type	<Integer>	Read-only	Configuration Type. Values: 0: IPv4, 1: IPv6.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/tacacs_server?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"tacacs_server": {

<b>"tacacs_key":<String_value></b>,

<b>"name":<String_value></b>,

<b>"ip_address":<String_value></b>,

"auth_timeout":<Integer_value>,

"id":<String_value>,

"port":<Integer_value>,

"accounting":<Boolean_value>,

"group_attr_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tacacs_server":[{

"tacacs_key":<String_value>,

"auth_timeout":<Integer_value>,

"name":<String_value>,

"group_attr_name":<String_value>,

"port":<Integer_value>,
```

```
"id":<String_value>,

"ip_address":<String_value>,

"accounting":<Boolean_value>,

"address_type":<Integer_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/tacacs_server/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/tacacs_server **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/tacacs_server **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of tacacs_server resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/tacacs_server **? pagesize=#no;pageno=#no** Use this query-parameter to get the tacacs_server resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/tacacs_server **?count=yes** Use this query-parameter to get the count of tacacs_server resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tacacs_server":[{

"tacacs_key":<String_value>,

"auth_timeout":<Integer_value>,

"name":<String_value>,

"group_attr_name":<String_value>,

"port":<Integer_value>,

"id":<String_value>,

"ip_address":<String_value>,

"accounting":<Boolean_value>,
```

```
"address_type":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/tacacs_server/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tacacs_server":[{

"tacacs_key":<String_value>,

"auth_timeout":<Integer_value>,

"name":<String_value>,

"group_attr_name":<String_value>,

"port":<Integer_value>,

"id":<String_value>,

"ip_address":<String_value>,
```

```
"accounting":<Boolean_value>,

"address_type":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/tacacs_server/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"tacacs_server":{

<b>"name":<String_value></b>,

<b>"id":<String_value></b>,

<b>"ip_address":<String_value></b>,

"auth_timeout":<Integer_value>,

"tacacs_key":<String_value>,

"port":<Integer_value>,

"accounting":<Boolean_value>,

"group_attr_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tacacs_server":[{

"tacacs_key":<String_value>,

"auth_timeout":<Integer_value>,

"name":<String_value>,

"group_attr_name":<String_value>,

"port":<Integer_value>,

"id":<String_value>,

"ip_address":<String_value>,

"accounting":<Boolean_value>,

"address_type":<Integer_value>}]}
```

task_command_log

Configuration for Task Command Log resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the task command logs.
status	<String>	Read-only	Status.
starttime	<Integer>	Read-only	Start Time.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
endtime	<Integer>	Read-only	End Time.
ip_address	<String>	Read-only	IP Address.
statusmessage	<String>	Read-only	Status Message.
command	<String>	Read-only	Command.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/task_command_log **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/task_command_log **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of task_command_log resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/task_command_log **? pagesize=#no;pageno=#no** Use this query-parameter to get the task_command_log resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/task_command_log **?count=yes** Use this query-parameter to get the count of task_command_log resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"task_command_log":[{

"protocol":<String_value>,

"status":<String_value>,

"index":<Integer_value>,

"starttime":<Integer_value>,

"id":<String_value>,

"task_device_id":<String_value>,

"endtime":<Integer_value>,

"ip_address":<String_value>,

"statusmessage":<String_value>,

"command":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/task_command_log/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"task_command_log":[{

"protocol":<String_value>,

"status":<String_value>,

"index":<Integer_value>,

"starttime":<Integer_value>,

"id":<String_value>,

"task_device_id":<String_value>,

"endtime":<Integer_value>,

"ip_address":<String_value>,

"statusmessage":<String_value>,

"command":<String_value>}]}
```

task_device_log

Configuration for Task Device Log resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the task device logs.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
status	<String>	Read-only	Status.
starttime	<Integer>	Read-only	Start Time.
device_user	<String>	Read-only	Device user whose credentials used for executing configuration job on the device..
ip_address	<String>	Read-only	IP Address.
endtime	<Integer>	Read-only	End Time.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/task_device_log **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/task_device_log **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of task_device_log resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/task_device_log **?pagesize=#no;pageno=#no** Use this query-parameter to get the task_device_log resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/task_device_log **?count=yes** Use this query-parameter to get the count of task_device_log resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"task_device_log":[{

"parent_name":<String_value>,

"status":<String_value>,

"task_id":<String_value>,

"parent_id":<String_value>,

"starttime":<Integer_value>,

"device_user":<String_value>,

"id":<String_value>,

"ip_address":<String_value>,

"endtime":<Integer_value>,

"cmds_completed_fraction":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/task_device_log/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"task_device_log":[{

"parent_name":<String_value>,

"status":<String_value>,

"task_id":<String_value>,

"parent_id":<String_value>,

"starttime":<Integer_value>,

"device_user":<String_value>,

"id":<String_value>,

"ip_address":<String_value>,

"endtime":<Integer_value>,

"cmds_completed_fraction":<String_value>}]}
```

task_log

Configuration for Task Log resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the task logs.
tag	<String>	Read-write	Tag. Used in tracking tasks created by specific entities, say an application.. Maximum length = 128
executed_by	<String>	Read-only	User Name.
status	<String>	Read-only	Status.
name	<String>	Read-only	Task Name.
starttime	<Integer>	Read-only	Start Time.
endtime	<Integer>	Read-only	End Time.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/task_log **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/task_log **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of task_log resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/task_log **? pagesize=#no;pageno=#no** Use this query-parameter to get the task_log resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/task_log **?count=yes** Use this query-parameter to get the count of task_log resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"task_log":[{

"executed_by":<String_value>,

"status":<String_value>,

"name":<String_value>,

"tenant_id":<String_value>,

"starttime":<Integer_value>,

"id":<String_value>,

"endtime":<Integer_value>,

"tag":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/task_log/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"task_log":[{

"executed_by":<String_value>,

"status":<String_value>,

"name":<String_value>,

"tenant_id":<String_value>,

"starttime":<Integer_value>,

"id":<String_value>,

"endtime":<Integer_value>,

"tag":<String_value>}]}
```

techsupport

Configuration for Technical Support resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
mode	<String>	Read-write	Technical support Mode, Possible values Appliance | XenServer | Management Service | Appliance_full | Instances. Minimum length = 1 Maximum length = 128
file_location_path	<String>	Read-write	File Location on Client for upload/ download.
file_name	<String>	Read-write	Technical support File Name. Minimum length = 1 Maximum length = 256
file_size	<Double>	Read-only	File Size.
file_last_modified	<String>	Read-only	Last Modified Time.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- START
- GET (ALL)
- GET
- DELETE
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

start

URL: https://<MGMT-IP>/nitro/v2/config/techsupport?action=start;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"techsupport": {

<b>"mode":<String_value></b>,

<b>"file_location_path":<String_value></b>,

<b>"file_name":<String_value></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"techsupport":[{

"enable_debug":<Boolean_value>,

"file_size":<Double_value>,

"mode":<String_value>,

"file_last_modified":<String_value>,

"enable_db_distribution":<Boolean_value>,

"file_location_path":<String_value>,

"file_name":<String_value>,

"act_id":<String_value>,

"vpx_list_for_techsupport":<String_value>,
```

```
"debug_duration":<Integer_value>,

"collect_options":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/techsupport **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/techsupport **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of techsupport resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/techsupport **? pagesize=#no;pageno=#no** Use this query-parameter to get the techsupport resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/techsupport **?count=yes** Use this query-parameter to get the count of techsupport resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"techsupport":[{

"enable_debug":<Boolean_value>,

"file_size":<Double_value>,

"mode":<String_value>,

"file_last_modified":<String_value>,

"enable_db_distribution":<Boolean_value>,

"file_location_path":<String_value>,

"file_name":<String_value>,

"act_id":<String_value>,

"vpx_list_for_techsupport":<String_value>,

"debug_duration":<Integer_value>,

"collect_options":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/techsupport/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"techsupport":[{

"enable_debug":<Boolean_value>,

"file_size":<Double_value>,

"mode":<String_value>,

"file_last_modified":<String_value>,

"enable_db_distribution":<Boolean_value>,

"file_location_path":<String_value>,

"file_name":<String_value>,

"act_id":<String_value>,

"vpx_list_for_techsupport":<String_value>,

"debug_duration":<Integer_value>,

"collect_options":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/techsupport/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

download

URL: https://<MGMT-IP>/nitro/v2/download/techsupport/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload: Binary Stream

tenant

Configuration for Tenants resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
system_resource	<tenant_system_resource>	Read-write	Tenant System Resource.
company_info	<tenant_company_info>	Read-write	Tenant Company Information.
parent_id	<String>	Read-write	Tenant ID of the parent Tenant..
access_to_parent	<Boolean>	Read-write	Enable Shell access for non-nsroot User(s).
password	<String>	Read-write	Password. Minimum length = 1 Maximum length = 128
id	<String>	Read-write	Id is system generated key for all the Tenants.
dbrole_name	<String>	Read-write	Database Role Name for tenant. Minimum length = 1 Maximum length = 128
service_type	<Double>	Read-write	Type of service, Setting the bits for Citrix ADM, NGS, WAF.
name	<String>	Read-write	Name of the Tenant. Minimum length = 1 Maximum length = 512

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
schema_name	<String>	Read-write	Schema Name for tenant. Minimum length = 1 Maximum length = 128
user_name	<String>	Read-write	User Name for tenant. Minimum length = 1 Maximum length = 128
preauthtoken	<String>	Read-write	Preauth token for a tenant. Minimum length = 1 Maximum length = 512
act_id	<String>	Read-write	Activity ID that is used by GUI to track the provisioning status..
entitlement	<String>	Read-write	Entitlement string received from citrix cloud..

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/tenant?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"tenant": {

"service_type":<Double_value>,

"dbrole_name":<String_value>,

"system_resource":<tenant_system_resource_value>,

"name":<String_value>,

"schema_name":<String_value>,

"company_info":<tenant_company_info_value>,

"user_name":<String_value>,

"parent_id":<String_value>,

"access_to_parent":<Boolean_value>,

"password":<String_value>,

"act_id":<String_value>,

"entitlement":<String_value>,

"preauthtoken":<String_value>,

"id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tenant":[{

"system_resource":<tenant_system_resource_value>,

"company_info":<tenant_company_info_value>,

"parent_id":<String_value>,

"access_to_parent":<Boolean_value>,

"password":<String_value>,

"id":<String_value>,

"cc_region":<String_value>,

"admin_tenant":<String_value>,

"dbrole_name":<String_value>,

"service_type":<Double_value>,

"is_tenant_post_54x":<Boolean_value>,

"name":<String_value>,

"schema_name":<String_value>,

"user_name":<String_value>,

"database_name":<String_value>,

"preauthtoken":<String_value>,

"act_id":<String_value>,

"entitlement":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/tenant/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/tenant **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/tenant **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of tenant resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/tenant **? pagesize=#no;pageno=#no** Use this query-parameter to get the tenant resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/tenant **?count=yes** Use this query-parameter to get the count of tenant resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tenant":[{

"system_resource":<tenant_system_resource_value>,

"company_info":<tenant_company_info_value>,

"parent_id":<String_value>,

"access_to_parent":<Boolean_value>,

"password":<String_value>,

"id":<String_value>,

"cc_region":<String_value>,

"admin_tenant":<String_value>,

"dbrole_name":<String_value>,

"service_type":<Double_value>,

"is_tenant_post_54x":<Boolean_value>,

"name":<String_value>,

"schema_name":<String_value>,

"user_name":<String_value>,

"database_name":<String_value>,
```

```
"preauthtoken":<String_value>,

"act_id":<String_value>,

"entitlement":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/tenant/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tenant":[{

"system_resource":<tenant_system_resource_value>,

"company_info":<tenant_company_info_value>,

"parent_id":<String_value>,

"access_to_parent":<Boolean_value>,

"password":<String_value>,
```

```
"id":<String_value>,

"cc_region":<String_value>,

"admin_tenant":<String_value>,

"dbrole_name":<String_value>,

"service_type":<Double_value>,

"is_tenant_post_54x":<Boolean_value>,

"name":<String_value>,

"schema_name":<String_value>,

"user_name":<String_value>,

"database_name":<String_value>,

"preauthtoken":<String_value>,

"act_id":<String_value>,

"entitlement":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/tenant/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"tenant":{

<b>"id":<String_value></b>,

"service_type":<Double_value>,

"dbrole_name":<String_value>,

"system_resource":<tenant_system_resource_value>,

"name":<String_value>,

"schema_name":<String_value>,

"company_info":<tenant_company_info_value>,

"user_name":<String_value>,

"parent_id":<String_value>,

"password":<String_value>,

"access_to_parent":<Boolean_value>,

"act_id":<String_value>,

"entitlement":<String_value>,

"preauthtoken":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tenant":[{

"system_resource":<tenant_system_resource_value>,
```

```
"company_info":<tenant_company_info_value>,

"parent_id":<String_value>,

"access_to_parent":<Boolean_value>,

"password":<String_value>,

"id":<String_value>,

"cc_region":<String_value>,

"admin_tenant":<String_value>,

"dbrole_name":<String_value>,

"service_type":<Double_value>,

"is_tenant_post_54x":<Boolean_value>,

"name":<String_value>,

"schema_name":<String_value>,

"user_name":<String_value>,

"database_name":<String_value>,

"preauthtoken":<String_value>,

"act_id":<String_value>,

"entitlement":<String_value>}]}
```

tenant_auth_server

Configuration for Authentication Server for the Tenant resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
tenant_id	<String>	Read-write	Id of the parent tenant.
parent_name	<String>	Read-write	.
auth_type	<String>	Read-write	Authentication Type.
server_details	<String>	Read-write	Department of the tenant.
id	<String>	Read-write	.
parent_id	<String>	Read-write	.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

tenant_company_info

Configuration for Company Information of Tenant resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
department	<String>	Read-write	Department of the tenant.
company_name	<String>	Read-write	Name of the tenant company.
zip_code	<String>	Read-write	Zipcode of the tenant.
state	<String>	Read-write	State of the tenant.
last_name	<String>	Read-write	URL of the tenant.
parent_id	<String>	Read-write	.
city	<String>	Read-write	City of the tenant.
tenant_id	<String>	Read-write	Id of the parent tenant.
url	<String>	Read-write	URL of the tenant.
address	<String>	Read-write	URL of the tenant.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	.
email_address	<String>	Read-write	Email Address of the tenant.
country	<String>	Read-write	Country of the tenant.
parent_name	<String>	Read-write	.
phone_number	<String>	Read-write	Phone number of the tenant.
preferred_lang	<String>	Read-write	Preferred Language.
first_name	<String>	Read-write	First name of the tenant.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

tenant_info

Configuration for Administrative Domain Information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ability_to_create_domain	<Boolean>	Read-write	Ability to create sub administrative domain.
level	<Integer>	Read-write	Level of Administrative Domain.
parent_name	<String>	Read-only	Parent Administrative Domain Name.
name	<String>	Read-only	Administrative Domain Name.
is_pooled_license	<Boolean>	Read-only	Platform is licensed for Triscaler feature.

Operations

(click to see Properties)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get

URL: https://<MGMT-IP>/nitro/v2/config/tenant_info **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tenant_info":[{

"ability_to_create_domain":<Boolean_value>,

"level":<Integer_value>,

"parent_name":<String_value>,

"name":<String_value>,

"is_pooled_license":<Boolean_value>}]}
```

tenant_network_pool

Configuration for Tenant Network Pools resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
tenant_id	<String>	Read-write	Id of the parent tenant.
parent_name	<String>	Read-write	.
id	<String>	Read-write	.
network_pool_id	<String>	Read-write	Id of network pool.
parent_id	<String>	Read-write	.
is_assigned	<Boolean>	Read-only	To check whether network pool is assigned or not.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

tenant_provisioning_status

Configuration for Activity Status resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
priority	<Integer>	Read-write	priority of the steps.
timestamp	<Double>	Read-write	timestamp in seconds.
status	<String>	Read-write	Status.
id	<String>	Read-write	Id is system generated key for all the task logs.
activity	<String>	Read-write	activity.
message	<String>	Read-write	Message.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- GET (ALL)

- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/tenant_provisioning_status **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/tenant_provisioning_status **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the

filtered set of tenant_provisioning_status resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/tenant_provisioning_status **?pagesize=#no;pageno=#no** Use this query-parameter to get the tenant_provisioning_status resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/tenant_provisioning_status **?count=yes** Use this query-parameter to get the count of tenant_provisioning_status resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tenant_provisioning_status":[{

"priority":<Integer_value>,

"timestamp":<Double_value>,

"act_id":<String_value>,

"status":<String_value>,

"id":<String_value>,
```

```
"activity":<String_value>,

"message":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/tenant_provisioning_status/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tenant_provisioning_status":[{

"priority":<Integer_value>,

"timestamp":<Double_value>,

"act_id":<String_value>,

"status":<String_value>,

"id":<String_value>,
```

```
"activity":<String_value>,

"message":<String_value>}]}
```

tenant_system_info

Configuration for This provides the FTU state of the user resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
services_subscribed	<String>	Read-write	Services Entitled By the user.
ftu_state	<Integer>	Read-write	Ftu state of the tenant.
username	<String>	Read-write	user logged in.
agent_download_state	<Integer>	Read-write	Agent download state.
services_subscribed_list_arr	<String[]>	Read-write	Subscribed Services list.
tenant_id	<String>	Read-only	Id of the parent tenant.
id	<String>	Read-only	Id is system generated key for all the syslog generic record..

Operations

(click to see Properties)

- ADD
- GET (ALL)

- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/tenant_system_info?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"tenant_system_info": {

"services_subscribed_list_arr":<String_value[]>,

"agent_download_state":<Integer_value>,

"username":<String_value>,

"ftu_state":<Integer_value>,

"services_subscribed":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tenant_system_info":[{

"services_subscribed":<String_value>,

"tenant_id":<String_value>,

"ftu_state":<Integer_value>,

"id":<String_value>,

"username":<String_value>,

"agent_download_state":<Integer_value>,
```

```
"services_subscribed_list_arr":<String_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/tenant_system_info **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/tenant_system_info **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of tenant_system_info resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/tenant_system_info **? pagesize=#no;pageno=#no** Use this query-parameter to get the tenant_system_info resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/tenant_system_info **?count=yes** Use this query-parameter to get the count of tenant_system_info resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tenant_system_info":[{

"services_subscribed":<String_value>,

"tenant_id":<String_value>,

"ftu_state":<Integer_value>,

"id":<String_value>,

"username":<String_value>,

"agent_download_state":<Integer_value>,

"services_subscribed_list_arr":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/tenant_system_info/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tenant_system_info":[{

"services_subscribed":<String_value>,

"tenant_id":<String_value>,

"ftu_state":<Integer_value>,

"id":<String_value>,

"username":<String_value>,

"agent_download_state":<Integer_value>,

"services_subscribed_list_arr":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/tenant_system_info/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"tenant_system_info":{

"services_subscribed_list_arr":<String_value[]>,

"agent_download_state":<Integer_value>,

"username":<String_value>,

"ftu_state":<Integer_value>,
```

```
"services_subscribed":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"tenant_system_info":[{

"services_subscribed":<String_value>,

"tenant_id":<String_value>,

"ftu_state":<Integer_value>,

"id":<String_value>,

"username":<String_value>,

"agent_download_state":<Integer_value>,

"services_subscribed_list_arr":<String_value>}]}
```

tenant_system_resource

Configuration for System Resources available for a tenant resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
available_disk_size	<Double>	Read-write	Available disk size.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
max_plt_bw	<Integer>	Read-write	Max Allowed Platinum bandwidth Mbps.
max_cpu_cores	<Integer>	Read-write	Maximum number of CPU Cores. Minimum value = 1 Maximum value =
parent_id	<String>	Read-write	.
max_scu	<Integer>	Read-write	Available number of Symmetric Crypto Units.
tenant_id	<String>	Read-write	Id of the parent tenant.
max_disk_size	<Double>	Read-write	Maximum Disk Space.
max_instances	<Integer>	Read-write	Maximum number of Instances. Minimum value = 1 Maximum value =
id	<String>	Read-write	.
parent_name	<String>	Read-write	.
available_instances	<Integer>	Read-write	Available number of instances. Maximum value =
max_std_bw	<Integer>	Read-write	Max Allowed Standard bandwidth Mbps.
max_memory	<Double>	Read-write	Maximum Memory.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
max_ent_bw	<Integer>	Read-write	Max Allowed Enterprise bandwidth Mbps.
available_memory	<Double>	Read-write	Available memory.
max_ssl_cards	<Integer>	Read-write	Maximum number of SSL cores.
available_cpu_cores	<Integer>	Read-write	Available number of CPU Cores. Maximum value =
max_acu	<Integer>	Read-write	Available number of Asymmetric Crypto Units.
available_scu	<Integer>	Read-only	Available number of Symmetric Crypto Units.
max_pps	<String>	Read-only	Maximum PPS .
available_std_bw	<Integer>	Read-only	Available Standard bandwidth in Mbps.
available_pps	<String>	Read-only	Available PPS.
available_throughput	<String>	Read-only	Available Throughput in Mbps.
available_acu	<Integer>	Read-only	Available number of Asymmetric Crypto Units.
max_throughput	<String>	Read-only	Maximum Throughput in Mbps.
available_ent_bw	<Integer>	Read-only	Available Enterprise bandwidth in Mbps.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
available_ssl_cards	<Integer>	Read-only	Available number of SSL Cores.
available_plt_bw	<Integer>	Read-only	Available Platinum bandwidth in Mbps.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

timezone_enum

Configuration for Time zones resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
enum_value	<String>	Read-only	Value.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/timezone_enum **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/timezone_enum **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of timezone_enum resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/timezone_enum **? pagesize=#no;pageno=#no** Use this query-parameter to get the timezone_enum resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/timezone_enum **?count=yes** Use this query-parameter to get the count of timezone_enum resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"timezone_enum":[{

"enum_key":<Integer_value>,

"enum_value":<String_value>}]}
```

trap_forward_profile

Configuration for Trap Forward Profile resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
trap_destination_port	<String>	Read-write	SNMP port to forward trap. Minimum length = 1 Maximum length = 128
trap_community	<String>	Read-write	SNMP community name using which we want to forward the trap. Minimum length = 1 Maximum length = 128
id	<String>	Read-write	Id is system generated key for all the mail profile..

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
trap_destination	<String>	Read-write	SMTP server IP. Minimum length = 1 Maximum length = 128
profile_name	<String>	Read-write	Profile name for the mail setting.. Minimum length = 1 Maximum length = 128
tenant_id	<String>	Read-only	Tenant Id .

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of

creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/trap_forward_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"trap_forward_profile": {

<b>"trap_destination_port":<String_value></b>,

<b>"trap_community":<String_value></b>,
```

```
<b>"trap_destination":<String_value></b>,

<b>"profile_name":<String_value></b>,

"id":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"trap_forward_profile":[{

"trap_destination_port":<String_value>,

"tenant_id":<String_value>,

"trap_community":<String_value>,

"id":<String_value>,

"trap_destination":<String_value>,

"profile_name":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/trap_forward_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/trap_forward_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/trap_forward_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of trap_forward_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/trap_forward_profile **? pagesize=#no;pageno=#no** Use this query-parameter to get the trap_forward_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/trap_forward_profile **?count=yes** Use this query-parameter to get the count of trap_forward_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"trap_forward_profile":[{

"trap_destination_port":<String_value>,

"tenant_id":<String_value>,

"trap_community":<String_value>,

"id":<String_value>,

"trap_destination":<String_value>,

"profile_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/trap_forward_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"trap_forward_profile":[{

"trap_destination_port":<String_value>,

"tenant_id":<String_value>,

"trap_community":<String_value>,

"id":<String_value>,

"trap_destination":<String_value>,

"profile_name":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/trap_forward_profile/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"trap_forward_profile":{

<b>"id":<String_value></b>,

"trap_destination_port":<String_value>,

"trap_destination":<String_value>,

"trap_community":<String_value>,

"profile_name":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"trap_forward_profile":[{

"trap_destination_port":<String_value>,

"tenant_id":<String_value>,

"trap_community":<String_value>,

"id":<String_value>,

"trap_destination":<String_value>,

"profile_name":<String_value>}]}
```

trust_preauthtoken

Configuration for Preauth token required for Agent registration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
service_name	<String>	Read-write	This is the Service name. Minimum length = 1 Maximum length = 512
name	<String>	Read-write	Name of the Tenant.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 512
token	<String>	Read-write	Preauth token for a tenant. Minimum length = 1 Maximum length = 512
api_gateway_url	<String>	Read-write	This is the service API-GW URL. Minimum length = 1 Maximum length = 512

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/trust_preauthtoken **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/trust_preauthtoken **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of trust_preauthtoken resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/trust_preauthtoken **? pagesize=#no;pageno=#no** Use this query-parameter to get the trust_preauthtoken resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/trust_preauthtoken **?count=yes** Use this query-parameter to get the count of trust_preauthtoken resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"trust_preauthtoken":[{

"service_name":<String_value>,

"name":<String_value>,

"token":<String_value>,

"api_gateway_url":<String_value>}]}
```

user_lockout_policy

Configuration for User Lockout Policy configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
user_lockout_interval	<Integer>	Read-write	User lockout Interval in minutes. Minimum value = 1 Maximum value =
invalid_logins	<Integer>	Read-write	No of invalid logins for User lockout .

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum value = 3 Maximum value =
id	<String>	Read-write	Id is system generated key.
enable_user_lockout	<Boolean>	Read-write	Enable user User lockout feature.

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/user_lockout_policy **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/user_lockout_policy **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of user_lockout_policy resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/user_lockout_policy **? pagesize=#no;pageno=#no** Use this query-parameter to get the user_lockout_policy resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/user_lockout_policy **?count=yes** Use this query-parameter to get the count of user_lockout_policy resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"user_lockout_policy":[{

"user_lockout_interval":<Integer_value>,

"invalid_logins":<Integer_value>,

"id":<String_value>,

"enable_user_lockout":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/user_lockout_policy/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"user_lockout_policy":[{
```

```
"user_lockout_interval":<Integer_value>,

"invalid_logins":<Integer_value>,

"id":<String_value>,

"enable_user_lockout":<Boolean_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/user_lockout_policy/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"user_lockout_policy":{

"invalid_logins":<Integer_value>,

"id":<String_value>,

"enable_user_lockout":<Boolean_value>,

"user_lockout_interval":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"user_lockout_policy":[{

"user_lockout_interval":<Integer_value>,

"invalid_logins":<Integer_value>,

"id":<String_value>,

"enable_user_lockout":<Boolean_value>}]}
```

Websense-Protector

- websense
- websense_device_profile
- websense_init_config
- websense_upgrade
- websense_vpx_image

websense

Configuration for Websense Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	Image Name, This parameter is used while provisioning VM Instance with XVA image, template_name is given priority if provided along with image_name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 128
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.
vm_memory_total	<Double>	Read-write	Total Memory of VM Instance in MB. 2048MB, 5120MB.
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
throughput_limit	<Double>	Read-write	Throughput Limit in Mbps set for VM Instance.
vlan_id_0_1	<Integer>	Read-write	VLAN id for the management interface 0/1. Maximum value =
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
description	<String>	Read-write	Description of managed device. Minimum length = 1 Maximum length = 512
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
fips_partition_name	<String>	Read-write	FIPS Partition Name. Minimum length = 1 Maximum length = 128
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.
initial_configuration	<websense_init_config>	Read-write	initial configuration parameters of websense.
burst_priority	<Integer>	Read-write	Burst Priority of the VM Instance between 1 and 4.
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
is_managed	<Boolean>	Read-write	Is Managed.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device. Minimum length = 1 Maximum length = 128
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
network_interfaces	<network_interface[]>	Read-write	Network Interfaces.
last_updated_time	<Double>	Read-write	Last Updated Time.
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.
host_ip_address	<String>	Read-write	Host IPAddress where VM is provisioned. Minimum length = 1 Maximum length = 64
id	<String>	Read-write	Id is system generated key for all the managed devices.
ipv6_address	<String>	Read-write	IPv6 Address.
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
license_edition	<String>	Read-write	Edition of instance.
number_of_acu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
device_family	<String>	Read-write	Device Family. Minimum length = 1 Maximum length = 64
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..
la_mgmt	<Boolean>	Read-write	Bond consisting of management ports on VM Instance.
vlan_id_0_2	<Integer>	Read-write	VLAN id for the management interface 0/2. Maximum value =
number_of_ssl_cores	<Integer>	Read-write	Assign number of ssl virtual functions to VM Instance.
type	<String>	Read-write	Type of device, (Xen | NS). Minimum length = 1 Maximum length = 64
throughput	<Double>	Read-write	Assign throughput in Mbps to VM Instance.
template_name	<String>	Read-write	Template Name, This parameter is used while provisioning VM Instance with template,

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			template_name is given priority if provided along with image_name. Minimum length = 1 Maximum length = 128
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64
max_burst_throughput	<Double>	Read-write	Maximum burst throughput in Mbps of VM Instance.
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal network (not required to be set for SDWAN).
node_id	<String>	Read-write	Node identification of a device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
throughput_allocation_mode	<Integer>	Read-write	Throughput Allocation Mode: 0-Fixed, 1-Burst-able.
number_of_scu	<Integer>	Read-write	Assign number of asymmetric crypto units to VM Instance.
number_of_cores	<Integer>	Read-write	Number of cores that are assigned to VM Instance.
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be A-B, and for Cluster it will be CLIP. Minimum length = 1 Maximum length = 128
std_bw_total	<Integer>	Read-write	Standard Bandwidth.
domain_name	<String>	Read-write	Domain name of VM Device. Minimum length = 1 Maximum length = 128
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1 Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_1_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/2 on VM Instance. Maximum value =
vrid_list_ipv4_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV4 VMAC Generation.
vlan_1_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/7 on VM Instance. Maximum value =
vrid_list_ipv6_1_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/2 for IPV6 VMAC Generation.
vrid_list_ipv4_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV4 VMAC Generation.
vrid_list_ipv6_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 10/6 for IPV6 VMAC Generation.
if_10_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/4 on VM Instance.
vlan_10_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/5 on VM Instance. Maximum value =
vrid_list_ipv6_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV6 VMAC Generation.
vrid_list_ipv6_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV6 VMAC Generation.
if_1_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/5 on VM Instance.
receiveuntagged_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtReceive Untagged Packets on 1/4 on VM Instance.
vrid_list_ipv6_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV6 VMAC Generation.
vrid_list_ipv6_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV6 VMAC Generation.
if_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/3 on VM Instance.
vrid_list_ipv6_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/5 for IPV6 VMAC Generation.
receiveuntagged_10_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/4 on VM Instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vrid_list_ipv6_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV6 VMAC Generation.
vlan_10_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/6 on VM Instance. Maximum value =
if_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/5 on VM Instance.
is_new_crypto	<Boolean>	Read-write	True if number_of_acu/ number_of_scu are used, false if number_of_ssl_core s is used.
vrid_list_ipv6_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV6 VMAC Generation.
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 128
if_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/7 on VM Instance.
receiveuntagged_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/8 on VM Instance.
vrid_list_ipv6_1_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/4 for IPV6 VMAC Generation.
receiveuntagged_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/6 on VM Instance.
vlan_1_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/4 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
if_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/2 on VM Instance.
receiveuntagged_1_ 5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/5 on VM Instance.
receiveuntagged_10 _6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/6 on VM Instance.
receiveuntagged_10 _1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/1 on VM Instance.
vrid_list_ipv4_1_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/6 for IPV4 VMAC Generation.
vrid_list_ipv6_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 1/5 for IPV6 VMAC Generation.
vrid_list_ipv6_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV6 VMAC Generation.
vrid_list_ipv4_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV4 VMAC Generation.
receiveuntagged_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/2 on VM Instance.
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
vrid_list_ipv4_1_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/5 for IPV4 VMAC Generation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/7 on VM Instance.
vlan_10_4	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/4 on VM Instance. Maximum value =
if_1_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/7 on VM Instance.
receiveuntagged_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/1 on VM Instance.
vrid_list_ipv6_1_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/7 for IPV6 VMAC Generation.
if_10_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtNetwork 10/1 on VM Instance.
if_10_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/8 on VM Instance.
l2_enabled	<Boolean>	Read-write	L2mode status of VM Instance.
vlan_10_7	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/7 on VM Instance. Maximum value =
vlan_10_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/1 on VM Instance. Maximum value =
receiveuntagged_1_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/2 on VM Instance.
reboot_vm_on_cpu_change	<Boolean>	Read-write	Reboot VMs on CPU change during resource allocation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vlan_1_6	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/6 on VM Instance. Maximum value =
vrid_list_ipv4_1_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/3for IPV4 VMAC Generation.
receiveuntagged_10_7	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/7 on VM Instance.
vrid_list_ipv4_1_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/1 for IPV4 VMAC Generation.
vlan_1_1	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/1 on VM Instance. Maximum value =
vrid_list_ipv6_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			br;gtVRID List for Interface 10/7 for IPV6 VMAC Generation.
vlan_1_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/8 on VM Instance. Maximum value =
vlan_10_2	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/2 on VM Instance. Maximum value =
receiveuntagged_10_5	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 10/5 on VM Instance.
vrid_list_ipv4_10_3	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/3 for IPV4 VMAC Generation.
vrid_list_ipv4_10_6	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 10/6 for IPV4 VMAC Generation.
vrid_list_ipv4_1_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 1/8 for IPV4 VMAC Generation.
receiveuntagged_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/8 on VM Instance.
vrid_list_ipv4_10_4	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/4 for IPV4 VMAC Generation.
vlan_1_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/3 on VM Instance. Maximum value =
receiveuntagged_10_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			on 10/3 on VM Instance.
if_10_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/6 on VM Instance.
if_1_6	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/6 on VM Instance.
vlan_10_3	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/3 on VM Instance. Maximum value =
vlan_1_5	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 1/5 on VM Instance. Maximum value =
if_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/3 on VM Instance.
vrid_list_ipv4_10_5	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Interface 10/5 for IPV4 VMAC Generation.
if_1_4	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/4 on VM Instance.
vrid_list_ipv4_10_8	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/8 for IPV4 VMAC Generation.
if_1_1	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/1 on VM Instance.
vrid_list_ipv4_10_7	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/7 for IPV4 VMAC Generation.
vlan_10_8	<Integer>	Read-write	This property is deprecated by network_interfaces;lt br;gtVLAN for Network 10/8 on VM Instance. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
receiveuntagged_1_3	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtReceive Untagged Packets on 1/3 on VM Instance.
crypto_change_requires_reboot	<Boolean>	Read-write	True if the current changes made by user requires a reboot of the VM else false.
vrid_list_ipv4_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for IPV4 VMAC Generation.
if_1_8	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 1/8 on VM Instance.
vrid_list_ipv4_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV4 VMAC Generation.
vrid_list_ipv6_10_2	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/2 for

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			IPV6 VMAC Generation.
if_10_2	<Boolean>	Read-write	This property is deprecated by network_interfaces;lt br;gtNetwork 10/2 on VM Instance.
vrid_list_ipv6_10_1	<String[]>	Read-write	This property is deprecated by network_interfaces;lt br;gtVRID List for Interface 10/1 for IPV6 VMAC Generation.
sdxtools_http_port	<Integer>	Read-only	Http Port number to communicate to SDXTools.
instance_state	<String>	Read-only	State of device, UP only if device accessible.
reason	<String>	Read-only	Reason of failure for this managed device.
websense_cpu_usag e	<Double>	Read-only	CPU Usage (%) of Websense Instance.
virtual_functions	<String>	Read-only	Virtual Functions assigned to VM Instance depending on SDX platform.
disk_space	<Double>	Read-only	Show Disk Space (GB) available to VM Instance.
upsince	<String>	Read-only	Upsince of managed device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
domid	<Double>	Read-only	DOM Id assigned to VM Instance viewed in hypervisor.
cpu_core_pe	<String>	Read-only	Packet Engine cores assigned to VM Instance.
tenant_id	<String>	Read-only	Tenant ID.
preferred_cpu_socket	<String>	Read-only	Preferred CPU Socket.
device_uuid	<String>	Read-only	Device UUID.
do_config	<Boolean>	Read-only	Do default config for managed device.
version	<String>	Read-only	Device Version.
host_id	<String>	Read-only	Host ID.
has_sdxtools	<Boolean>	Read-only	True if SDX Tools are installed on this VM.
vm_memory_used	<Double>	Read-only	Memory Used (MB) by VM Instance.
pe_status	<String>	Read-only	Policy Engine status of Websense Instance.
number_of_cpu	<Integer>	Read-only	Number of CPU that is assigned to VM Instance.
is_swg	<Boolean>	Read-only	Boolean to indicate whether a VM is SWG.
sysid	<String>	Read-only	System ID.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
vm_state	<String>	Read-only	State of Virtual Machine (Running | Halted).
uuid	<String>	Read-only	UUID of VM Instance generated by Hypervisor.
pama_status	<String>	Read-only	PAMA status of Websense Instance.
cpu_core_mgmt	<String>	Read-only	Management CPU cores assigned to VM Instance.
ssl_virtual_functions	<String>	Read-only	SSL Virtual Functions assigned to VM Instance.
uptime	<String>	Read-only	Uptime of device.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.
assigned_cpu_socket	<String>	Read-only	Assigned CPU Socket 0 or 1.
sdxtools_public_key	<String>	Read-only	SDXTools HTTPS Public Key.
status	<String>	Read-only	Status of managed device.
websense_memory_usage	<Double>	Read-only	Memory Usage (%) of Websense Instance.
vm_description	<String>	Read-only	Description of the VM instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
systemname	<String>	Read-only	Device System Name.
mac_eth0	<String>	Read-only	MAC Address of eth0 on VM Instance.
vm_rx	<Double>	Read-only	In Throughput of VM Instance in Mbps.
disk_allocation	<String>	Read-only	Disk allocation for VM Instance.
vm_tx	<Double>	Read-only	Out Throughput of VM Instance in Mbps.
vm_memory_free	<Double>	Read-only	Free Memory (MB) available in VM Instance.
vm_memory_usage	<Double>	Read-only	Memory Usage (%) of VM Instance.
sdxtools_version	<String>	Read-only	SDXTools version running on VM Instance.
vm_cpu_usage	<Double>	Read-only	CPU Usage (%) of VM Instance.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- REBOOT
- STOP
- ADD
- FORCE_REBOOT

- GET (ALL)
- GET
- DELETE
- UPDATE
- FORCE_STOP
- START

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

reboot

URL: https://<MGMT-IP>/nitro/v2/config/websense/id_value<String>?action=reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"websense": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

stop

URL: https://<MGMT-IP>/nitro/v2/config/websense/id_value<String>?action=stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"websense": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

add

URL: https://<MGMT-IP>/nitro/v2/config/websense?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"websense": {

<b>"ip_address":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,
```

```
"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,

"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,
```

```
"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,
```

```
"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,

"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,
```

```
"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"initial_configuration":<websense_init_config_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,
```

```
"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"is_member_ifc":<Boolean_value>,

"vrid_list_ipv4":<String_value>,

"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"id":<String_value>,

"l2_enabled":<Boolean_value>,

"interface_name":<String_value>,
```

```
"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,
```

```
"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"max_burst_throughput":<Double_value>,

"if_1_3":<Boolean_value>,

"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,
```

```
"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"websense":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,
```

```
"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"websense_cpu_usage":<Double_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,
```

```
"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"initial_configuration":<websense_init_config_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"pe_status":<String_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,
```

```
"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,
```

```
"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"pama_status":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,
```

```
"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,
```

```
"status":<String_value>,

"gateway":<String_value>,

"websense_memory_usage":<Double_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,
```

```
"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,
```

```
"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,
```

```
"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,
```

```
"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,
```

```
"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

force_reboot

URL: https://<MGMT-IP>/nitro/v2/config/websense/id_value<String>?action=force_reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"websense": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/websense **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/websense **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of websense resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/websense **? pagesize=#no;pageno=#no** Use this query-parameter to get the websense resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/websense **?count=yes** Use this query-parameter to get the count of websense resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"websense":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,
```

```
"websense_cpu_usage":<Double_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"initial_configuration":<websense_init_config_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,
```

```
"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,

"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"pe_status":<String_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,
```

```
"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,
```

```
"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>

"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"pama_status":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,
```

```
"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,

"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"websense_memory_usage":<Double_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,
```

```
"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,

"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,
```

```
"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,

"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,
```

```
"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,

"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,
```

```
"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,
```

```
"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,

"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,
```

```
"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,

"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/websense/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"websense":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"websense_cpu_usage":<Double_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,
```

```
"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"initial_configuration":<websense_init_config_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,
```

```
"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"pe_status":<String_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,
```

```
"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>
```

```
"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"pama_status":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,
```

```
"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"websense_memory_usage":<Double_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,
```

```
"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,
```

```
"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,
```

```
"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,
```

```
"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,
```

```
"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,
```

```
"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/websense/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

update

URL: https://<MGMT-IP>/nitro/v2/config/websense/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"websense":{

<b>"id":<String_value></b>,

"vlan_1_2":<Integer_value>,

"std_bw_config":<Integer_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"vlan_id_0_1":<Integer_value>,

"if_10_4":<Boolean_value>,

"name":<String_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value[]>,

"vrid_list_ipv6_1_1":<String_value[]>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"vrid_list_ipv6_10_5":<String_value[]>,

"vrid_list_ipv6_1_8":<String_value[]>,

"vcpu_config":<Integer_value>,

"netmask":<String_value>,

"ent_bw_config":<Integer_value>,

"vrid_list_ipv6_1_6":<String_value[]>,
```

```
"profile_username":<String_value>,

"datacenter_id":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"instance_mode":<String_value>,

"vrid_list_ipv6_1_4":<String_value[]>,

"vlan_1_4":<Integer_value>,

"std_bw_available":<Integer_value>,

"vrid_list_ipv4_1_7":<String_value[]>,

"vrid_list_ipv6_10_8":<String_value[]>,

"internal_ip_address":<String_value>,

"receiveuntagged_10_2":<Boolean_value>,

"vrid_list_ipv4_1_5":<String_value[]>,

"profile_password":<String_value>,

"receiveuntagged_1_1":<Boolean_value>,

"if_1_7":<Boolean_value>,

"vrid_list_ipv6_1_7":<String_value[]>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"receiveuntagged_1_2":<Boolean_value>,

"vlan_10_1":<Integer_value>,

"vlan_10_7":<Integer_value>,

"ipv6_address":<String_value>,
```

```
"mgmt_ip_address":<String_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"device_family":<String_value>,

"vrid_list_ipv6_10_7":<String_value[]>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_1_8":<Boolean_value>,

"vrid_list_ipv4_10_6":<String_value[]>,

"vrid_list_ipv4_10_4":<String_value[]>,

"type":<String_value>,

"throughput":<Double_value>,

"if_1_6":<Boolean_value>,

"template_name":<String_value>,

"gateway":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value[]>,

"if_1_1":<Boolean_value>,

"vlan_10_8":<Integer_value>,

"number_of_scu":<Integer_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value[]>,
```

```
"domain_name":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value[]>,

"vrid_list_ipv4_1_2":<String_value[]>,

"image_name":<String_value>,

"vlan_1_7":<Integer_value>,

"hostname":<String_value>,

"vrid_list_ipv6_1_2":<String_value[]>,

"vm_memory_total":<Double_value>,

"vrid_list_ipv4_1_4":<String_value[]>,

"vrid_list_ipv6_10_6":<String_value[]>,

"ent_bw_available":<Integer_value>,

"description":<String_value>,

"fips_partition_name":<String_value>,

"vrid_list_ipv6_10_4":<String_value[]>,

"vrid_list_ipv6_1_3":<String_value[]>,

"if_10_3":<Boolean_value>,

"receiveuntagged_10_4":<Boolean_value>,

"initial_configuration":<websense_init_config_value>,

"vlan_10_6":<Integer_value>,

"burst_priority":<Integer_value>,

"if_10_5":<Boolean_value>,
```

```
"is_new_crypto":<Boolean_value>,

"if_10_7":<Boolean_value>,

"is_managed":<Boolean_value>,

"receiveuntagged_1_6":<Boolean_value>,

"nexthop_v6":<String_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"ipv4_address":<String_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"profile_name":<String_value>,

"vrid_list_ipv4_1_6":<String_value[]>,

"vrid_list_ipv6_1_5":<String_value[]>,

"network_interfaces":[{

<b>"port_name":<String_value></b>,

"name_server":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"gateway":<String_value>,

"vrid_list_ipv6":<String_value>,

"parent_id":<String_value>,

"is_member_ifc":<Boolean_value>,

"vrid_list_ipv4":<String_value>,
```

```
"mac_address":<String_value>,

"netmask":<String_value>,

"ip_address":<String_value>,

"id":<String_value>,

"l2_enabled":<Boolean_value>,

"interface_name":<String_value>,

"parent_name":<String_value>,

"vlan_whitelist_array":<String_value[]>,

"mac_mode":<String_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"vrid_list_ipv4_array":<String_value[]>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vrid_list_ipv6_array":<String_value[]>,

"is_vlan_applied":<Boolean_value>,

"vlan_whitelist":<String_value>}],

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"last_updated_time":<Double_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,
```

```
"l2_enabled":<Boolean_value>,

"license_edition":<String_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value[]>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value[]>,

"vlan_1_1":<Integer_value>,

"license_grace_time":<Integer_value>,

"vlan_1_8":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value[]>,

"la_mgmt":<Boolean_value>,

"vrid_list_ipv4_1_8":<String_value[]>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"max_burst_throughput":<Double_value>,

"if_1_3":<Boolean_value>,
```

```
"config_type":<Integer_value>,

"vrid_list_ipv4_10_5":<String_value[]>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"ip_address":<String_value>,

"vrid_list_ipv4_10_7":<String_value[]>,

"throughput_allocation_mode":<Integer_value>,

"number_of_cores":<Integer_value>,

"receiveuntagged_1_3":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value[]>,

"std_bw_total":<Integer_value>,

"display_name":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv6_10_2":<String_value[]>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"websense":[{

"image_name":<String_value>,

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"std_bw_config":<Integer_value>,

"vm_memory_total":<Double_value>,

"is_internal_intf_added":<Boolean_value>,

"gateway_ipv6":<String_value>,

"throughput_limit":<Double_value>,

"sdxtools_http_port":<Integer_value>,

"instance_state":<String_value>,

"vlan_id_0_1":<Integer_value>,

"vnc_console_url":<String_value>,

"provision_with_xva":<Boolean_value>,

"name":<String_value>,

"reason":<String_value>,

"websense_cpu_usage":<Double_value>,

"virtual_functions":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,
```

```
"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"fips_partition_name":<String_value>,

"domid":<Double_value>,

"cpu_core_pe":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"ent_bw_total":<Integer_value>,

"host_type":<String_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"initial_configuration":<websense_init_config_value>,

"preferred_cpu_socket":<String_value>,

"burst_priority":<Integer_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"has_sdxtools":<Boolean_value>,
```

```
"is_managed":<Boolean_value>,

"old_max_burst_throughput":<Double_value>,

"instance_mode":<String_value>,

"vm_memory_used":<Double_value>,

"pe_status":<String_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"ipv4_address":<String_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"is_swg":<Boolean_value>,

"internal_ip_address":<String_value>,

"dns":<String_value>,

"sysid":<String_value>,

"network_interfaces":[{

"device_channel_name":<String_value>,

"is_mgmt_ifc":<Boolean_value>,

"name_server":<String_value>,

"vrid_list_ipv6":<String_value>,

"gateway":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,
```

```
"xen_vf_index":<Integer_value>,

"parent_id":<String_value>,

"vrid_list_ipv4":<String_value>,

"is_member_ifc":<Boolean_value>,

"mac_address":<String_value>,

"parent_channel_id":<String_value>,

"channel_type":<String_value>,

"l2_enabled":<Boolean_value>,

"id":<String_value>,

"ip_address":<String_value>,

"netmask":<String_value>,

"port_name":<String_value>,

"parent_name":<String_value>,

"channel_timeout":<String_value>,

"mac_mode":<String_value>,

"channel_ha_monitoring":<Boolean_value>,

"managed_device_id":<String_value>,

"vlan":<Integer_value>,

"receiveuntagged":<Boolean_value>,

"sdx_formation_network_id":<String_value>,

"vlan_whitelist":<String_value>

"interface_name":<String_value>
```

```
"vrid_list_ipv4_array":<String_value>

"vrid_list_ipv6_array":<String_value>

"vlan_whitelist_array":<String_value>

"is_vlan_applied":<Boolean_value>}],

"vm_state":<String_value>,

"uuid":<String_value>,

"pama_status":<String_value>,

"last_updated_time":<Double_value>,

"cpu_core_mgmt":<String_value>,

"ssl_virtual_functions":<String_value>,

"old_throughput":<Double_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"number_of_acu":<Integer_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"device_family":<String_value>,
```

```
"license_grace_time":<Integer_value>,

"la_mgmt":<Boolean_value>,

"assigned_cpu_socket":<String_value>,

"vlan_id_0_2":<Integer_value>,

"number_of_ssl_cores":<Integer_value>,

"type":<String_value>,

"throughput":<Double_value>,

"sdxtools_public_key":<String_value>,

"template_name":<String_value>,

"status":<String_value>,

"gateway":<String_value>,

"websense_memory_usage":<Double_value>,

"vm_description":<String_value>,

"max_burst_throughput":<Double_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"mac_eth0":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"vm_rx":<Double_value>,

"node_id":<String_value>,

"privStr":<String_value>,

"disk_allocation":<String_value>,
```

```
"vm_tx":<Double_value>,

"ip_address":<String_value>,

"throughput_allocation_mode":<Integer_value>,

"number_of_scu":<Integer_value>,

"ping_state":<Integer_value>,

"number_of_cores":<Integer_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"domain_name":<String_value>,

"nexthop":<String_value>,

"vm_memory_free":<Double_value>,

"plt_bw_config":<Integer_value>,

"formation_instance_id":<String_value>,

"vm_memory_usage":<Double_value>,

"sdxtools_version":<String_value>,

"vm_cpu_usage":<Double_value>,

"parent_inventory":<Boolean_value>,

"vlan_1_2":<Integer_value>,

"vrid_list_ipv4_1_2":<String_value>,

"vlan_1_7":<Integer_value>,

"vrid_list_ipv6_1_2":<String_value>,

"vrid_list_ipv4_1_4":<String_value>,
```

```
"act_id":<String_value>,

"vrid_list_ipv6_10_6":<String_value>,

"if_10_4":<Boolean_value>,

"vlan_10_5":<Integer_value>,

"vrid_list_ipv6_10_3":<String_value>,

"vrid_list_ipv6_1_1":<String_value>,

"if_1_5":<Boolean_value>,

"receiveuntagged_1_4":<Boolean_value>,

"vrid_list_ipv6_10_4":<String_value>,

"vrid_list_ipv6_1_3":<String_value>,

"if_10_3":<Boolean_value>,

"xva_desc_present":<Boolean_value>,

"vrid_list_ipv6_10_5":<String_value>,

"receiveuntagged_10_4":<Boolean_value>,

"advertise_vm_uuid":<String_value>,

"vrid_list_ipv6_1_8":<String_value>,

"vlan_10_6":<Integer_value>,

"if_10_5":<Boolean_value>,

"is_new_crypto":<Boolean_value>,

"delete_template_name":<String_value>,

"vrid_list_ipv6_1_6":<String_value>,

"profile_username":<String_value>,
```

```
"if_10_7":<Boolean_value>,

"default_password":<String_value>,

"receiveuntagged_10_8":<Boolean_value>,

"vrid_list_ipv6_1_4":<String_value>,

"receiveuntagged_1_6":<Boolean_value>,

"vlan_1_4":<Integer_value>,

"if_1_2":<Boolean_value>,

"receiveuntagged_1_5":<Boolean_value>,

"receiveuntagged_10_6":<Boolean_value>,

"receiveuntagged_10_1":<Boolean_value>,

"vrid_list_ipv4_1_6":<String_value>,

"vrid_list_ipv6_1_5":<String_value>,

"vrid_list_ipv6_10_8":<String_value>,

"vrid_list_ipv4_1_7":<String_value>,

"make_vm_ready":<Boolean_value>,

"receiveuntagged_10_2":<Boolean_value>,

"profile_password":<String_value>,

"vrid_list_ipv4_1_5":<String_value>,

"receiveuntagged_1_7":<Boolean_value>,

"vlan_10_4":<Integer_value>,

"if_1_7":<Boolean_value>,

"receiveuntagged_1_1":<Boolean_value>,
```

```
"vrid_list_ipv6_1_7":<String_value>,

"if_10_1":<Boolean_value>,

"if_10_8":<Boolean_value>,

"l2_enabled":<Boolean_value>,

"vlan_10_7":<Integer_value>,

"vlan_10_1":<Integer_value>,

"receiveuntagged_1_2":<Boolean_value>,

"reboot_vm_on_cpu_change":<Boolean_value>,

"vlan_1_6":<Integer_value>,

"vrid_list_ipv4_1_3":<String_value>,

"receiveuntagged_10_7":<Boolean_value>,

"vrid_list_ipv4_1_1":<String_value>,

"vlan_1_1":<Integer_value>,

"vrid_list_ipv6_10_7":<String_value>,

"vlan_1_8":<Integer_value>,

"vlan_10_2":<Integer_value>,

"receiveuntagged_10_5":<Boolean_value>,

"vrid_list_ipv4_10_3":<String_value>,

"vrid_list_ipv4_10_6":<String_value>,

"vrid_list_ipv4_1_8":<String_value>,

"receiveuntagged_1_8":<Boolean_value>,

"sync_operation":<Boolean_value>,
```

```
"additional_info":<String_value>,

"vrid_list_ipv4_10_4":<String_value>,

"vlan_1_3":<Integer_value>,

"receiveuntagged_10_3":<Boolean_value>,

"if_10_6":<Boolean_value>,

"if_1_6":<Boolean_value>,

"vlan_10_3":<Integer_value>,

"vlan_1_5":<Integer_value>,

"if_1_3":<Boolean_value>,

"vpx_image_dir":<String_value>,

"vrid_list_ipv4_10_5":<String_value>,

"if_1_4":<Boolean_value>,

"vrid_list_ipv4_10_8":<String_value>,

"if_1_1":<Boolean_value>,

"vrid_list_ipv4_10_7":<String_value>,

"vlan_10_8":<Integer_value>,

"convert_template_name":<String_value>,

"receiveuntagged_1_3":<Boolean_value>,

"crypto_change_requires_reboot":<Boolean_value>,

"vrid_list_ipv4_10_2":<String_value>,

"if_1_8":<Boolean_value>,

"vrid_list_ipv4_10_1":<String_value>,
```

```
"vrid_list_ipv6_10_2":<String_value>,

"if_10_2":<Boolean_value>,

"vrid_list_ipv6_10_1":<String_value>}]}
```

force_stop

URL: https://<MGMT-IP>/nitro/v2/config/websense/id_value<String>?action=force_stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"websense": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

start

URL: https://<MGMT-IP>/nitro/v2/config/websense/id_value<String>?action=start;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"websense": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

websense_device_profile

Configuration for Websense device profile resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Profile Name. Minimum length = 1 Maximum length = 128
svm_ns_comm	<String>	Read-write	Communication protocol (http or https) with Instances.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 10
use_global_setting_for_communication_with_ns	<Boolean>	Read-write	True, if the communication with Instance needs to be global and not device specific.
id	<String>	Read-write	Id is system generated key for all the device profiles.
type	<String>	Read-write	Profile Type, This must be with in specified supported instance types: blx,sdvanvw,ns,nssdx,cbwanopt,cpx. Minimum length = 1 Maximum length = 128
ns_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix ADC profiles.
password	<String>	Read-write	Instance credentials.Password for this profile. Minimum length = 1 Maximum length = 127
snmpsecuritylevel	<String>	Read-write	SNMP v3 security level for this profile.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
username	<String>	Read-write	Instance credentials.Username provided in the profile will be used to contact the instance. Minimum length = 1 Maximum length = 127
snmpauthprotocol	<String>	Read-write	SNMP v3 auth protocol for this profile.
ssh_port	<String>	Read-write	SSH port to connect to the device.
snmpprivprotocol	<String>	Read-write	SNMP v3 priv protocol for this profile.
host_password	<String>	Read-write	Host Password for this profile.Used for BLX form factor of ADC. Minimum length = 1 Maximum length = 127
snmpversion	<String>	Read-write	SNMP version for this profile.
passphrase	<String>	Read-write	Passphrase with which private key is encrypted.
host_username	<String>	Read-write	Host User Name for this profile.Used for BLX form factor of ADC.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 127
snmpsecurityname	<String>	Read-write	SNMP v3 security name for this profile. Maximum length = 31
ssl_private_key	<String>	Read-write	SSL Private Key for key based authentication.
ssl_cert	<String>	Read-write	SSL Certificate for certificate based authentication.
http_port	<Integer>	Read-write	HTTP port to connect to the device.
snmpcommunity	<String>	Read-write	SNMP community for this profile. Maximum length = 31
https_port	<Integer>	Read-write	HTTPS port to connect to the device.
max_wait_time_reboot	<String>	Read-write	Max waiting time to reboot Citrix ADC.
snmpprivpassword	<String>	Read-write	SNMP v3 priv password for this profile. Minimum length = 8 Maximum length = 31

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cb_profile_name	<String>	Read-write	Profile Name, This is one of the already created Citrix SD-WAN profiles.
snmpauthpassword	<String>	Read-write	SNMP v3 auth password for this profile. Minimum length = 8 Maximum length = 31
tenant_id	<String>	Read-only	Tenant Id of device profile.
is_default	<Boolean>	Read-only	True, if this profile is system generated and can not be deleted.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/websense_device_profile?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"websense_device_profile": {

<b>"name":<String_value></b>,
```

"password":<String_value>,

"username":<String_value>,

"snmpsecuritylevel":<String_value>,

"svm_ns_comm":<String_value>,

"ssh_port":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"type":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"ns_profile_name":<String_value>,

"snmpauthprotocol":<String_value>,

"snmpprivprotocol":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

```
"snmpsecurityname":<String_value>,

"snmpcommunity":<String_value>,

"snmpauthpassword":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"websense_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,
```

```
"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/websense_device_profile/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/websense_device_profile **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/websense_device_profile **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of websense_device_profile resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/websense_device_profile **?pagesize=#no;pageno=#no** Use this query-parameter to get the websense_device_profile resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/websense_device_profile **?count=yes** Use this query-parameter to get the count of websense_device_profile resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"websense_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,
```

```
"ssl_private_key":<String_value>,

"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/websense_device_profile/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"websense_device_profile":[{

"name":<String_value>,

"svm_ns_comm":<String_value>,

"tenant_id":<String_value>,

"use_global_setting_for_communication_with_ns":<Boolean_value>,

"id":<String_value>,

"is_default":<Boolean_value>,

"type":<String_value>,

"ns_profile_name":<String_value>,

"password":<String_value>,

"snmpsecuritylevel":<String_value>,

"username":<String_value>,

"snmpauthprotocol":<String_value>,

"ssh_port":<String_value>,

"snmpprivprotocol":<String_value>,

"host_password":<String_value>,

"snmpversion":<String_value>,

"passphrase":<String_value>,

"host_username":<String_value>,

"snmpsecurityname":<String_value>,

"ssl_private_key":<String_value>,
```

```
"ssl_cert":<String_value>,

"http_port":<Integer_value>,

"snmpcommunity":<String_value>,

"https_port":<Integer_value>,

"max_wait_time_reboot":<String_value>,

"snmpprivpassword":<String_value>,

"cb_profile_name":<String_value>,

"snmpauthpassword":<String_value>}]}
```

websense_init_config

Configuration for Initial Configuration Parameters of websense resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
parent_name	<String>	Read-write	.
parent_id	<String>	Read-write	.
dss_passwd	<String>	Read-write	Login Credentials of DSS Manager.
dss_addr	<String>	Read-write	Address of DSS Manager.
root_passwd	<String>	Read-write	Root Profile Password of Websense Protector instance.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
dss_admin	<String>	Read-write	Login Credentials of DSS Manager.
time_zone	<String>	Read-write	Time Zone of Websense Protector instance.
id	<String>	Read-write	.
admin_passwd	<String>	Read-write	Admin profile Password of Websense Protector instance.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

websense_upgrade

Configuration for Upgrade WEBSENSE Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
websense_ip_address_arr	<String[]>	Read-write	List of WEBSENSE IP Address. Minimum length = 1 Maximum length = 64
image_name	<String>	Read-write	WEBSENSE Image Name. Minimum length = 1 Maximum length = 128
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- UPGRADE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

upgrade

URL: https://<MGMT-IP>/nitro/v2/config/websense_upgrade?action=upgrade;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"websense_upgrade": {

<b>"websense_ip_address_arr":<String_value[]></b>,

<b>"image_name":<String_value></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"websense_upgrade":[{

"act_id":<String_value>,

"websense_ip_address_arr":<String_value>,

"image_name":<String_value>}]}
```

websense_vpx_image

Configuration for XVA images of Websense Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/websense_vpx_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/websense_vpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/websense_vpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of websense_vpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/websense_vpx_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the websense_vpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/websense_vpx_image **?count=yes** Use this query-parameter to get the count of websense_vpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"websense_vpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,
```

```
"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/websense_vpx_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"websense_vpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,
```

```
"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/websense_vpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

XenServer

- cb_loopback
- channel
- channel_info
- clean_install
- disk_adapter
- fips_partition
- ftw_events

- host_cpu_core
- host_interface
- hostcpu
- logical_disk
- physical_disk
- snmp_interface
- virtual_nmi
- xen
- xen_applied_hotfix
- xen_bluecatvpx_image
- xen_brvpx_image
- xen_cb_host_interface
- xen_core
- xen_cpu_layout
- xen_cpu_switches
- xen_health_disk
- xen_health_ftw
- xen_health_hsm
- xen_health_interface
- xen_health_monitor
- xen_health_monitor_fan_speed
- xen_health_monitor_misc
- xen_health_monitor_temp
- xen_health_monitor_voltage
- xen_health_resource
- xen_health_resource_hw
- xen_health_resource_sw
- xen_health_sr
- xen_health_ssl_core
- xen_health_summary

- xen_hotfix
- xen_hsm
- xen_hsm_fw_image
- xen_hyper_thread
- xen_image
- xen_lacp_member_stats
- xen_lacp_stats
- xen_nsvpx_image
- xen_panwvpx_image
- xen_smx_gsbvpx_image
- xen_supplemental_pack
- xen_trend_microvpx_image
- xen_upgrade
- xen_websensevpx_image

cb_loopback

Configuration for Loopback Settings resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
loopback_enabled	<Boolean>	Read-write	Loopback Settings.
id	<String>	Read-write	Id is system generated key.

Operations

(click to see Properties)

- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/cb_loopback **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/cb_loopback **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of cb_loopback resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/cb_loopback **?pagesize=#no;pageno=#no** Use this query-parameter to get the cb_loopback resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/cb_loopback **?count=yes** Use this query-parameter to get the count of cb_loopback resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cb_loopback":[{

"loopback_enabled":<Boolean_value>,

"id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/cb_loopback/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cb_loopback":[{

"loopback_enabled":<Boolean_value>,

"id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/cb_loopback/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"cb_loopback":{

"id":<String_value>,

"loopback_enabled":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"cb_loopback":[{

"loopback_enabled":<Boolean_value>,

"id":<String_value>}]}
```

channel

Configuration for Channels on the system resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
channel_tag_all_vlans	<Boolean>	Read-write	If true then all the member interfaces of this channel are tagged. Possible values: true and false.
lacp_channel_time	<String>	Read-write	LACP time. Possible values: SHORT and LONG. Minimum length = 4 Maximum length = 10

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
channel_interface_list	<String[]>	Read-write	Comma separated list of interfaces that are part of this channel if this interface represents a channel (10/1, 10/4). Maximum length = 256
channel_alias	<String>	Read-write	Alias name for this channel. Minimum length = 4 Maximum length = 31
channel_type	<String>	Read-write	Channel type if this interface represents a channel (LACP or Static). Minimum length = 4 Maximum length = 16
mtu	<Integer>	Read-write	MTU value, should be between 1500-9126.
channel_id	<String>	Read-write	Channel ID if this interface represents a channel (LA/1, LA/2 ...). Minimum length = 4 Maximum length = 5

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
static_channel_state	<Boolean>	Read-write	Static channel state (Enabled/Disabled).
channel_ha_monitoring	<Boolean>	Read-write	HA-monitoring control for the channel. Possible values: true and false.
sync_operation	<Boolean>	Read-write	sync operation.
channel_throughput	<Integer>	Read-write	Minimum required throughput in Mbps for this channel. Maximum value =
channel_bandwidth_normal	<Integer>	Read-write	Lower end threshold of the channel bandwidth usage in Mbps. Maximum value =
channel_bandwidth_high	<Integer>	Read-write	Higher end threshold of the channel bandwidth usage in Mbps. Maximum value =
state	<String>	Read-only	State of the port..

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/channel?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"channel": {

<b>"channel_interface_list":<String_value[]></b>,

<b>"channel_type":<String_value></b>,

<b>"channel_id":<String_value></b>,

<b>"static_channel_state":<Boolean_value></b>,

"channel_tag_all_vlans":<Boolean_value>,

"lacp_channel_time":<String_value>,

"channel_alias":<String_value>,

"mtu":<Integer_value>,

"channel_ha_monitoring":<Boolean_value>,

"sync_operation":<Boolean_value>,

"channel_throughput":<Integer_value>,

"channel_bandwidth_high":<Integer_value>,

"channel_bandwidth_normal":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"channel":[{
```

```
"channel_tag_all_vlans":<Boolean_value>,

"state":<String_value>,

"lacp_channel_time":<String_value>,

"host_type":<String_value>,

"channel_interface_list":<String_value>,

"channel_alias":<String_value>,

"host_ip_address":<String_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_id":<String_value>,

"static_channel_state":<Boolean_value>,

"channel_ha_monitoring":<Boolean_value>,

"sync_operation":<Boolean_value>,

"channel_throughput":<Integer_value>,

"channel_bandwidth_normal":<Integer_value>,

"channel_bandwidth_high":<Integer_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/channel?action=delete;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"channel": {

<b>"channel_id":<String_value></b>,

"channel_tag_all_vlans":<Boolean_value>,

"lacp_channel_time":<String_value>,

"channel_interface_list":<String_value[]>,

"channel_alias":<String_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_ha_monitoring":<Boolean_value>,

"static_channel_state":<Boolean_value>,

"sync_operation":<Boolean_value>,

"channel_throughput":<Integer_value>,

"channel_bandwidth_high":<Integer_value>,

"channel_bandwidth_normal":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"channel":[{

"channel_tag_all_vlans":<Boolean_value>,

"state":<String_value>,

"lacp_channel_time":<String_value>,

"host_type":<String_value>,

"channel_interface_list":<String_value>,

"channel_alias":<String_value>,

"host_ip_address":<String_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_id":<String_value>,

"static_channel_state":<Boolean_value>,

"channel_ha_monitoring":<Boolean_value>,

"sync_operation":<Boolean_value>,

"channel_throughput":<Integer_value>,

"channel_bandwidth_normal":<Integer_value>,

"channel_bandwidth_high":<Integer_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/channel **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/channel **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of channel resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/channel **?pagesize=#no;pageno=#no** Use this query-parameter to get the channel resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/channel **?count=yes** Use this query-parameter to get the count of channel resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"channel":[{

"channel_tag_all_vlans":<Boolean_value>,

"state":<String_value>,

"lacp_channel_time":<String_value>,

"host_type":<String_value>,

"channel_interface_list":<String_value>,

"channel_alias":<String_value>,

"host_ip_address":<String_value>,
```

```
"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_id":<String_value>,

"static_channel_state":<Boolean_value>,

"channel_ha_monitoring":<Boolean_value>,

"sync_operation":<Boolean_value>,

"channel_throughput":<Integer_value>,

"channel_bandwidth_normal":<Integer_value>,

"channel_bandwidth_high":<Integer_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/channel/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"channel":{

<b>"channel_id":<String_value></b>,

"channel_tag_all_vlans":<Boolean_value>,

"lacp_channel_time":<String_value>,
```

```
"channel_interface_list":<String_value[]>,

"channel_alias":<String_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_ha_monitoring":<Boolean_value>,

"static_channel_state":<Boolean_value>,

"sync_operation":<Boolean_value>,

"channel_throughput":<Integer_value>,

"channel_bandwidth_high":<Integer_value>,

"channel_bandwidth_normal":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"channel":[{

"channel_tag_all_vlans":<Boolean_value>,

"state":<String_value>,

"lacp_channel_time":<String_value>,

"host_type":<String_value>,

"channel_interface_list":<String_value>,

"channel_alias":<String_value>,
```

```
"host_ip_address":<String_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_id":<String_value>,

"static_channel_state":<Boolean_value>,

"channel_ha_monitoring":<Boolean_value>,

"sync_operation":<Boolean_value>,

"channel_throughput":<Integer_value>,

"channel_bandwidth_normal":<Integer_value>,

"channel_bandwidth_high":<Integer_value>}]}
```

channel_info

Configuration for Provides platfor specific channel information resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
management_interfaces_list	<String[]>	Read-only	list of management interfaces that are available to be used at present on this system.
all_channel_ids	<String[]>	Read-only	list of channels ids that are possible on this system.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
available_channel_ids	<String[]>	Read-only	list of channels ids that are available to be used at present on this system.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/channel_info **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/channel_info **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of channel_info resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/channel_info **? pagesize=#no;pageno=#no** Use this query-parameter to get the channel_info resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/channel_info **?count=yes** Use this query-parameter to get the count of channel_info resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"channel_info":[{
```

```
"management_interfaces_list":<String_value>,

"host_type":<String_value>,

"all_channel_ids":<String_value>,

"host_ip_address":<String_value>,

"available_channel_ids":<String_value>}]}
```

clean_install

Configuration for Xen Upgrade resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	image_name.
is_supported	<Boolean>	Read-only	Is Clean-Install supported.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- ADD
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/clean_install?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"clean_install": {

<b>"image_name":<String_value></b>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"clean_install":[{

"is_supported":<Boolean_value>,

"act_id":<String_value>,

"image_name":<String_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/clean_install **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/clean_install **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of clean_install resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/clean_install **? pagesize=#no;pageno=#no** Use this query-parameter to get the clean_install resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/clean_install **?count=yes** Use this query-parameter to get the count of clean_install resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"clean_install":[{

"is_supported":<Boolean_value>,

"act_id":<String_value>,

"image_name":<String_value>}]}
```

disk_adapter

Configuration for XenServer Disk Adapaters resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key .
pdcount	<Integer>	Read-only	Physical Disk Count for this Adapter.
adapter_id	<Integer>	Read-only	Adapter ID.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ldcount	<Integer>	Read-only	Logical Disk Count for this Adapter.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/disk_adapter **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/disk_adapter **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of disk_adapter resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/disk_adapter **? pagesize=#no;pageno=#no** Use this query-parameter to get the disk_adapter resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/disk_adapter **?count=yes** Use this query-parameter to get the count of disk_adapter resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"disk_adapter":[{
```

```
"pdcount":<Integer_value>,

"adapter_id":<Integer_value>,

"ldcount":<Integer_value>,

"id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/disk_adapter/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"disk_adapter":[{

"pdcount":<Integer_value>,

"adapter_id":<Integer_value>,

"ldcount":<Integer_value>,

"id":<String_value>}]}
```

fips_partition

Configuration for HSM FIPS Partition resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
contexts_size	<Double>	Read-write	Maximum contexts for the partition. Minimum value = 4 Maximum value =
pco_password	<String>	Read-write	Password for specified user on Citrix ADC Instance. Minimum length = 1 Maximum length = 127
key_store_size	<Double>	Read-write	No of keys for the partition. Minimum value = 4 Maximum value =
partition_name	<String>	Read-write	FIPS Partition Name. Minimum length = 1 Maximum length = 15
crypto_core_capacity	<Integer>	Read-write	Max Crypto Cores for the partition. Minimum value = 1 Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
pco_username	<String>	Read-write	User name to be configured on partition. Minimum length = 1 Maximum length = 127
id	<String>	Read-write	Id is system generated key for all the FIPS partitions.
instance_name	<String>	Read-only	Associated VPX(if any) of the partition.
partition_state	<String>	Read-only	Partition state: Initialized, Zeroized etc.
host_ip_address	<String>	Read-only	host_ip_address.
available_key_store_size	<Double>	Read-only	Available no of keys for the partition.
partition_id	<String>	Read-only	Id is HSM generated key for all fips partitions.

Operations

(click to see Properties)

- ADD
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/fips_partition?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"fips_partition": {

<b>"contexts_size":<Double_value></b>,

<b>"key_store_size":<Double_value></b>,

<b>"partition_name":<String_value></b>,

<b>"crypto_core_capacity":<Integer_value></b>,

"id":<String_value>,

"pco_password":<String_value>,

"pco_username":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"fips_partition":[{

"instance_name":<String_value>,

"contexts_size":<Double_value>,

"partition_state":<String_value>,

"pco_password":<String_value>,

"key_store_size":<Double_value>,

"partition_name":<String_value>,
```

```
"crypto_core_capacity":<Integer_value>,

"host_ip_address":<String_value>,

"pco_username":<String_value>,

"available_key_store_size":<Double_value>,

"id":<String_value>,

"partition_id":<String_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/fips_partition/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/fips_partition **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/fips_partition **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of fips_partition resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/fips_partition **?pagesize=#no;pageno=#no** Use this query-parameter to get the fips_partition resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/fips_partition **?count=yes** Use this query-parameter to get the count of fips_partition resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"fips_partition":[{

"instance_name":<String_value>,

"contexts_size":<Double_value>,

"partition_state":<String_value>,

"pco_password":<String_value>,

"key_store_size":<Double_value>,

"partition_name":<String_value>,
```

```
"crypto_core_capacity":<Integer_value>,

"host_ip_address":<String_value>,

"pco_username":<String_value>,

"available_key_store_size":<Double_value>,

"id":<String_value>,

"partition_id":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/fips_partition/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"fips_partition":[{

"instance_name":<String_value>,

"contexts_size":<Double_value>,
```

```
"partition_state":<String_value>,

"pco_password":<String_value>,

"key_store_size":<Double_value>,

"partition_name":<String_value>,

"crypto_core_capacity":<Integer_value>,

"host_ip_address":<String_value>,

"pco_username":<String_value>,

"available_key_store_size":<Double_value>,

"id":<String_value>,

"partition_id":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/fips_partition/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"fips_partition":{

<b>"id":<String_value></b>,

"contexts_size":<Double_value>,
```

```
"key_store_size":<Double_value>,

"pco_password":<String_value>,

"crypto_core_capacity":<Integer_value>,

"partition_name":<String_value>,

"pco_username":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"fips_partition":[{

"instance_name":<String_value>,

"contexts_size":<Double_value>,

"partition_state":<String_value>,

"pco_password":<String_value>,

"key_store_size":<Double_value>,

"partition_name":<String_value>,

"crypto_core_capacity":<Integer_value>,

"host_ip_address":<String_value>,

"pco_username":<String_value>,

"available_key_store_size":<Double_value>,

"id":<String_value>,
```

```
"partition_id":<String_value>}]}
```

ftw_events

Configuration for FTW Events resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
response	<String>	Read-only	Response.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/ftw_events **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/ftw_events **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of ftw_events resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/ftw_events **? pagesize=#no;pageno=#no** Use this query-parameter to get the ftw_events resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/ftw_events **?count=yes** Use this query-parameter to get the count of ftw_events resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"ftw_events":[{

"response":<String_value>}]}
```

host_cpu_core

Configuration for Host CPU Cores resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
instances	<String>	Read-only	VM Instances running on this Core.
cpu	<Integer>	Read-only	Physical CPU to which the Core belongs to (0 or 1).
individual_thread_usage	<String>	Read-only	Individual Thread Usage (%).
core_number	<Integer>	Read-only	Core Number.
avg_usage	<Double>	Read-only	Average CPU Usage (%).
hyper_threads	<String>	Read-only	Hyper-threads assoicated with this Core.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/host_cpu_core **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/host_cpu_core **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of host_cpu_core resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/host_cpu_core **?pagesize=#no;pageno=#no** Use this query-parameter to get the host_cpu_core resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/host_cpu_core **?count=yes** Use this query-parameter to get the count of host_cpu_core resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"host_cpu_core":[{

"instances":<String_value>,

"cpu":<Integer_value>,

"individual_thread_usage":<String_value>,

"core_number":<Integer_value>,

"avg_usage":<Double_value>,

"hyper_threads":<String_value>}]}
```

host_interface

Configuration for XenServer Interface resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
auto_applied	<Boolean>	Read-write	True if AUTO is selected as speed.
range	<String>	Read-write	Range for Base Mac Address. Minimum length = 1 Maximum length = 50
channel_tag_all_vlans	<Boolean>	Read-write	If true then all the member interfaces of this channel are tagged. Possible values: true and false.
mapped_port	<String>	Read-write	Mapped Port Name Ex: eth0. Minimum length = 1 Maximum length = 50
lacp_channel_time	<String>	Read-write	LACP time. Possible values: fast and slow. Minimum length = 4 Maximum length = 10

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
channel_interface_list	<String>	Read-write	Comma separated list of interfaces that are part of this channel if this interface represents a channel (10/1, 10/4). Maximum length = 256
channel_alias	<String>	Read-write	Alias name for this channel. Minimum length = 4 Maximum length = 31
id	<String>	Read-write	Id is system generated key .
add_mac_address	<Boolean>	Read-write	Add Mac Address.
apply_mac_address	<Boolean>	Read-write	Apply Mac Address.
channel_type	<String>	Read-write	Channel type if this interface represents a channel (LACP or Static). Minimum length = 4 Maximum length = 16
mtu	<Integer>	Read-write	MTU value, should be between 1500-9126.
channel_id	<String>	Read-write	Channel ID if this interface represents

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			a channel (LA/1, LA/2 ...). Minimum length = 4 Maximum length = 5
display_name	<String>	Read-write	Display Name Ex: Management Port. Minimum length = 1 Maximum length = 50
port	<String>	Read-write	Port Name Ex: 10/1. Minimum length = 1 Maximum length = 50
static_channel_state	<Boolean>	Read-write	Static channel state (Enabled/Disabled).
channel_ha_monitoring	<Boolean>	Read-write	HA-monitoring control for the channel. Possible values: true and false.
base_mac_address	<String>	Read-write	Mac Address. Minimum length = 1 Maximum length = 32
channel_throughput	<Integer>	Read-write	Minimum required throughput in Mbps for this channel. Maximum value =

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
channel_bandwidth_high	<Integer>	Read-write	Higher end threshold of the channel bandwidth usage in Mbps. Maximum value =
channel_bandwidth_normal	<Integer>	Read-write	Lower end threshold of the channel bandwidth usage in Mbps. Maximum value =
flow_control_tx	<Boolean>	Read-write	TX Pause.
vlan_strippping_state	<Integer>	Read-write	VLAN Stripping State, 0: Disabled 1: Enabled.
speed	<String>	Read-write	Actual speed . Minimum length = 1 Maximum length = 128
support_auto_neg	<Boolean>	Read-write	true if the port supports auto-negotiation.
flow_control_auto_neg	<Boolean>	Read-write	Auto Negotiation For Flow Control.
duplex	<String>	Read-write	Duplex. Minimum length = 1 Maximum length = 128
sync_operation	<Boolean>	Read-write	sync operation.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
adv_auto_neg	<Boolean>	Read-write	true if the advertised auto-negotiation for the port is true.
flow_control_rx	<Boolean>	Read-write	RX Pause.
admin_state	<String>	Read-only	Admin can enable or disable the interface.
xen_bond_uuid	<String>	Read-only	UUID of the corresponding bond in Xenserver.
is_mgmt_bond	<Boolean>	Read-only	True if this bond is on management interface.
lacp_channel_mac_address	<String>	Read-only	Mac address of the bond on Xenserver.
interface_type	<String>	Read-only	Indicates if this is an interface or a channel or a member interface of a channel.
supported_link_modes	<String[]>	Read-only	Supported Link Modes.
state	<String>	Read-only	State of the port..
device_name	<String>	Read-only	Device Name.
act_id	<String>	Read-only	Activity Id.
port_type	<String>	Read-only	Port Type.
supported_port_types	<String[]>	Read-only	Supported Port Types.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cpu_socket	<String>	Read-only	CPU Socket to which this interface belong to.
adv_link_modes	<String[]>	Read-only	Advertised Link Modes.

Operations

(click to see Properties)

- ADD
- RESET
- DELETE
- GET (ALL)
- GET
- UPDATE

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/host_interface?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"host_interface": {

<b>"channel_interface_list":<String_value></b>,

<b>"channel_type":<String_value></b>,

<b>"channel_id":<String_value></b>,

"auto_applied":<Boolean_value>,

"flow_control_tx":<Boolean_value>,

"range":<String_value>,

"mapped_port":<String_value>,
```

```
"channel_tag_all_vlans":<Boolean_value>,

"lacp_channel_time":<String_value>,

"channel_alias":<String_value>,

"speed":<String_value>,

"vlan_strippping_state":<Integer_value>,

"support_auto_neg":<Boolean_value>,

"apply_mac_address":<Boolean_value>,

"id":<String_value>,

"add_mac_address":<Boolean_value>,

"mtu":<Integer_value>,

"flow_control_auto_neg":<Boolean_value>,

"display_name":<String_value>,

"port":<String_value>,

"static_channel_state":<Boolean_value>,

"channel_ha_monitoring":<Boolean_value>,

"duplex":<String_value>,

"base_mac_address":<String_value>,

"sync_operation":<Boolean_value>,

"adv_auto_neg":<Boolean_value>,

"channel_throughput":<Integer_value>,

"flow_control_rx":<Boolean_value>,

"channel_bandwidth_normal":<Integer_value>,
```

```
"channel_bandwidth_high":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"host_interface":[{

"auto_applied":<Boolean_value>,

"range":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"mapped_port":<String_value>,

"admin_state":<String_value>,

"lacp_channel_time":<String_value>,

"xen_bond_uuid":<String_value>,

"host_type":<String_value>,

"channel_interface_list":<String_value>,

"channel_alias":<String_value>,

"is_mgmt_bond":<Boolean_value>,

"host_ip_address":<String_value>,

"lacp_channel_mac_address":<String_value>,

"id":<String_value>,

"add_mac_address":<Boolean_value>,
```

```
"apply_mac_address":<Boolean_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_id":<String_value>,

"display_name":<String_value>,

"port":<String_value>,

"static_channel_state":<Boolean_value>,

"channel_ha_monitoring":<Boolean_value>,

"base_mac_address":<String_value>,

"interface_type":<String_value>,

"channel_throughput":<Integer_value>,

"channel_bandwidth_high":<Integer_value>,

"channel_bandwidth_normal":<Integer_value>,

"flow_control_tx":<Boolean_value>,

"flow_control_changed":<Boolean_value>,

"supported_link_modes":<String_value>,

"state":<String_value>,

"device_name":<String_value>,

"act_id":<String_value>,

"vlan_strippping_state":<Integer_value>,

"master_pif_uuid":<String_value>,

"speed":<String_value>,
```

```
"link_speed_changed":<Boolean_value>,

"support_auto_neg":<Boolean_value>,

"port_type":<String_value>,

"supported_port_types":<String_value>,

"primary_slave_uuid":<String_value>,

"flow_control_auto_neg":<Boolean_value>,

"cpu_socket":<String_value>,

"duplex":<String_value>,

"sync_operation":<Boolean_value>,

"adv_auto_neg":<Boolean_value>,

"flow_control_rx":<Boolean_value>,

"adv_link_modes":<String_value>}]}
```

reset

URL: https://<MGMT-IP>/nitro/v2/config/host_interface/id_value<String>?action=reset;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"host_interface": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/host_interface/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/host_interface **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/host_interface **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of host_interface resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/host_interface **?pagesize=#no;pageno=#no** Use this query-parameter to get the host_interface resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/host_interface **?count=yes** Use this query-parameter to get the count of host_interface resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"host_interface":[{

"auto_applied":<Boolean_value>,

"range":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"mapped_port":<String_value>,

"admin_state":<String_value>,

"lacp_channel_time":<String_value>,
```

```
"xen_bond_uuid":<String_value>,

"host_type":<String_value>,

"channel_interface_list":<String_value>,

"channel_alias":<String_value>,

"is_mgmt_bond":<Boolean_value>,

"host_ip_address":<String_value>,

"lacp_channel_mac_address":<String_value>,

"id":<String_value>,

"add_mac_address":<Boolean_value>,

"apply_mac_address":<Boolean_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_id":<String_value>,

"display_name":<String_value>,

"port":<String_value>,

"static_channel_state":<Boolean_value>,

"channel_ha_monitoring":<Boolean_value>,

"base_mac_address":<String_value>,

"interface_type":<String_value>,

"channel_throughput":<Integer_value>,

"channel_bandwidth_high":<Integer_value>,

"channel_bandwidth_normal":<Integer_value>,
```

```
"flow_control_tx":<Boolean_value>,

"flow_control_changed":<Boolean_value>,

"supported_link_modes":<String_value>,

"state":<String_value>,

"device_name":<String_value>,

"act_id":<String_value>,

"vlan_strippping_state":<Integer_value>,

"master_pif_uuid":<String_value>,

"speed":<String_value>,

"link_speed_changed":<Boolean_value>,

"support_auto_neg":<Boolean_value>,

"port_type":<String_value>,

"supported_port_types":<String_value>,

"primary_slave_uuid":<String_value>,

"flow_control_auto_neg":<Boolean_value>,

"cpu_socket":<String_value>,

"duplex":<String_value>,

"sync_operation":<Boolean_value>,

"adv_auto_neg":<Boolean_value>,

"flow_control_rx":<Boolean_value>,

"adv_link_modes":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/host_interface/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"host_interface":[{

"auto_applied":<Boolean_value>,

"range":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"mapped_port":<String_value>,

"admin_state":<String_value>,

"lacp_channel_time":<String_value>,

"xen_bond_uuid":<String_value>,

"host_type":<String_value>,
```

```
"channel_interface_list":<String_value>,

"channel_alias":<String_value>,

"is_mgmt_bond":<Boolean_value>,

"host_ip_address":<String_value>,

"lacp_channel_mac_address":<String_value>,

"id":<String_value>,

"add_mac_address":<Boolean_value>,

"apply_mac_address":<Boolean_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_id":<String_value>,

"display_name":<String_value>,

"port":<String_value>,

"static_channel_state":<Boolean_value>,

"channel_ha_monitoring":<Boolean_value>,

"base_mac_address":<String_value>,

"interface_type":<String_value>,

"channel_throughput":<Integer_value>,

"channel_bandwidth_high":<Integer_value>,

"channel_bandwidth_normal":<Integer_value>,

"flow_control_tx":<Boolean_value>,

"flow_control_changed":<Boolean_value>,
```

```
"supported_link_modes":<String_value>,

"state":<String_value>,

"device_name":<String_value>,

"act_id":<String_value>,

"vlan_strippping_state":<Integer_value>,

"master_pif_uuid":<String_value>,

"speed":<String_value>,

"link_speed_changed":<Boolean_value>,

"support_auto_neg":<Boolean_value>,

"port_type":<String_value>,

"supported_port_types":<String_value>,

"primary_slave_uuid":<String_value>,

"flow_control_auto_neg":<Boolean_value>,

"cpu_socket":<String_value>,

"duplex":<String_value>,

"sync_operation":<Boolean_value>,

"adv_auto_neg":<Boolean_value>,

"flow_control_rx":<Boolean_value>,

"adv_link_modes":<String_value>}]}
```

update

URL: https://<MGMT-IP>/nitro/v2/config/host_interface/id_value<String> **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"host_interface":{

<b>"mapped_port":<String_value></b>,

<b>"id":<String_value></b>,

"auto_applied":<Boolean_value>,

"flow_control_tx":<Boolean_value>,

"range":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"lacp_channel_time":<String_value>,

"channel_interface_list":<String_value>,

"channel_alias":<String_value>,

"speed":<String_value>,

"vlan_strippping_state":<Integer_value>,

"support_auto_neg":<Boolean_value>,

"channel_type":<String_value>,

"add_mac_address":<Boolean_value>,

"apply_mac_address":<Boolean_value>,
```

```
"mtu":<Integer_value>,

"channel_id":<String_value>,

"flow_control_auto_neg":<Boolean_value>,

"display_name":<String_value>,

"port":<String_value>,

"static_channel_state":<Boolean_value>,

"channel_ha_monitoring":<Boolean_value>,

"duplex":<String_value>,

"base_mac_address":<String_value>,

"sync_operation":<Boolean_value>,

"adv_auto_neg":<Boolean_value>,

"channel_throughput":<Integer_value>,

"flow_control_rx":<Boolean_value>,

"channel_bandwidth_normal":<Integer_value>,

"channel_bandwidth_high":<Integer_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"host_interface":[{

"auto_applied":<Boolean_value>,
```

```
"range":<String_value>,

"channel_tag_all_vlans":<Boolean_value>,

"mapped_port":<String_value>,

"admin_state":<String_value>,

"lacp_channel_time":<String_value>,

"xen_bond_uuid":<String_value>,

"host_type":<String_value>,

"channel_interface_list":<String_value>,

"channel_alias":<String_value>,

"is_mgmt_bond":<Boolean_value>,

"host_ip_address":<String_value>,

"lacp_channel_mac_address":<String_value>,

"id":<String_value>,

"add_mac_address":<Boolean_value>,

"apply_mac_address":<Boolean_value>,

"channel_type":<String_value>,

"mtu":<Integer_value>,

"channel_id":<String_value>,

"display_name":<String_value>,

"port":<String_value>,

"static_channel_state":<Boolean_value>,

"channel_ha_monitoring":<Boolean_value>,
```

```
"base_mac_address":<String_value>,

"interface_type":<String_value>,

"channel_throughput":<Integer_value>,

"channel_bandwidth_high":<Integer_value>,

"channel_bandwidth_normal":<Integer_value>,

"flow_control_tx":<Boolean_value>,

"flow_control_changed":<Boolean_value>,

"supported_link_modes":<String_value>,

"state":<String_value>,

"device_name":<String_value>,

"act_id":<String_value>,

"vlan_strippping_state":<Integer_value>,

"master_pif_uuid":<String_value>,

"speed":<String_value>,

"link_speed_changed":<Boolean_value>,

"support_auto_neg":<Boolean_value>,

"port_type":<String_value>,

"supported_port_types":<String_value>,

"primary_slave_uuid":<String_value>,

"flow_control_auto_neg":<Boolean_value>,

"cpu_socket":<String_value>,

"duplex":<String_value>,
```

```
"sync_operation":<Boolean_value>,

"adv_auto_neg":<Boolean_value>,

"flow_control_rx":<Boolean_value>,

"adv_link_modes":<String_value>}]}
```

hostcpu

Configuration for Host CPU resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the CPUs.
features	<String>	Read-only	CPU features.
flags	<String>	Read-only	CPU Flags.
number	<Integer>	Read-only	CPU number.
model	<Integer>	Read-only	CPU Model.
uuid	<String>	Read-only	UUID of this CPU.
speed	<Integer>	Read-only	CPU speed.
cpu_usage	<Double>	Read-only	CPU Usage (%).
stepping	<String>	Read-only	CPU Stepping.
family	<Integer>	Read-only	CPU family.
vendor	<String>	Read-only	CPU Vendor.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
modelname	<String>	Read-only	CPU Model Name.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/hostcpu **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/hostcpu **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of hostcpu resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/hostcpu **? pagesize=#no;pageno=#no** Use this query-parameter to get the hostcpu resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/hostcpu **?count=yes** Use this query-parameter to get the count of hostcpu resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"hostcpu":[{

"features":<String_value>,

"flags":<String_value>,
```

```
"number":<Integer_value>,

"model":<Integer_value>,

"uuid":<String_value>,

"host_type":<String_value>,

"speed":<Integer_value>,

"host_ip_address":<String_value>,

"cpu_usage":<Double_value>,

"id":<String_value>,

"stepping":<String_value>,

"family":<Integer_value>,

"vendor":<String_value>,

"modelname":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/hostcpu/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"hostcpu":[{

"features":<String_value>,

"flags":<String_value>,

"number":<Integer_value>,

"model":<Integer_value>,

"uuid":<String_value>,

"host_type":<String_value>,

"speed":<Integer_value>,

"host_ip_address":<String_value>,

"cpu_usage":<Double_value>,

"id":<String_value>,

"stepping":<String_value>,

"family":<Integer_value>,

"vendor":<String_value>,

"modelname":<String_value>}]}
```

logical_disk

Configuration for Raid Logical Drive resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
host_ip_address	<String>	Read-write	host_ip_address. Minimum length = 1 Maximum length = 128
id	<String>	Read-write	Id is system generated key for all logical drives.
physical_disk_slot_2	<String>	Read-write	Second Slot for Raid Logical Drive.
physical_disk_slot_1	<String>	Read-write	First Slot for Raid Logical Drive.
force_clean	<Boolean>	Read-write	Force Clean Physical Disks before creating logical drive.
virtualdrive	<Integer>	Read-only	Virtual Drive.
name	<String>	Read-only	Logical Drive Name.
state	<String>	Read-only	State.
size	<String>	Read-only	Logical Drive Size.
adapter_id	<Integer>	Read-only	Adapter ID.
drives	<String>	Read-only	Drives.
sr_uuid	<String>	Read-only	SR UUID.
targetid	<Integer>	Read-only	Target ID.

Operations

(click to see Properties)

- ADD
- DELETE

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/logical_disk?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"logical_disk": {

"host_ip_address":<String_value>,

"id":<String_value>,

"physical_disk_slot_2":<String_value>,

"physical_disk_slot_1":<String_value>,

"force_clean":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"logical_disk":[{

"virtualdrive":<Integer_value>,

"name":<String_value>,

"state":<String_value>,

"size":<String_value>,

"adapter_id":<Integer_value>,

"drives":<String_value>,
```

```
"sr_uuid":<String_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"targetid":<Integer_value>,

"physical_disk_slot_2":<String_value>,

"physical_disk_slot_1":<String_value>,

"response":<String_value>,

"force_clean":<Boolean_value>}]}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/logical_disk/id_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/logical_disk **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/logical_disk **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of logical_disk resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/logical_disk **? pagesize=#no;pageno=#no** Use this query-parameter to get the logical_disk resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/logical_disk **?count=yes** Use this query-parameter to get the count of logical_disk resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"logical_disk":[{

"virtualdrive":<Integer_value>,

"name":<String_value>,

"state":<String_value>,

"size":<String_value>,
```

```
"adapter_id":<Integer_value>,

"drives":<String_value>,

"sr_uuid":<String_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"targetid":<Integer_value>,

"physical_disk_slot_2":<String_value>,

"physical_disk_slot_1":<String_value>,

"response":<String_value>,

"force_clean":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/logical_disk/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"logical_disk":[{

"virtualdrive":<Integer_value>,

"name":<String_value>,

"state":<String_value>,

"size":<String_value>,

"adapter_id":<Integer_value>,

"drives":<String_value>,

"sr_uuid":<String_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"targetid":<Integer_value>,

"physical_disk_slot_2":<String_value>,

"physical_disk_slot_1":<String_value>,

"response":<String_value>,

"force_clean":<Boolean_value>}]}
```

physical_disk

Configuration for Raid Physical Drive resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
logical_disk_name	<String>	Read-write	Logical Disk Name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 128
host_ip_address	<String>	Read-write	host_ip_address. Minimum length = 1 Maximum length = 128
id	<String>	Read-write	Id is system generated key for all physical drives.
slot	<Integer>	Read-write	Slot/Bay Number.
force_clean	<Boolean>	Read-write	Force Clean Disk before Rebuilding.
ld_number	<Integer>	Read-only	Logical Drive number.
deviceid	<Integer>	Read-only	Device ID.
size	<String>	Read-only	Disk Size.
firmwarestate	<String>	Read-only	Firmware State.
adapter_id	<Integer>	Read-only	Adapter ID.
mec	<String>	Read-only	mec.
disk_index	<Integer>	Read-only	Disk Index.
oec	<String>	Read-only	Drives.
is_used	<Boolean>	Read-only	Is Used.
enclosure	<Integer>	Read-only	Enclosure.
inquiry	<String>	Read-only	Inquiry.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
wwn	<String>	Read-only	wwn.
disk_name	<String>	Read-only	Disk Name.
physicalstate	<String>	Read-only	Physical State: Present OR Not Present.
foreignstate	<String>	Read-only	Foreign State.
mediatype	<String>	Read-only	Media Type.
partner_state	<String>	Read-only	Partner Disk firmware state.

Operations

(click to see Properties)

- MAKE_GOOD_PD
- REPLACE_MISSING
- REMOVE_PD
- LOCATE_PD_START
- GET (ALL)
- GET
- LOCATE_PD_STOP

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

make_good_pd

URL: https://<MGMT-IP>/nitro/v2/config/physical_disk?action=make_good_pd;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"physical_disk": {

"logical_disk_name":<String_value>,
```

```
"host_ip_address":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,

"force_clean":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"physical_disk":[{

"ld_number":<Integer_value>,

"deviceid":<Integer_value>,

"logical_disk_name":<String_value>,

"size":<String_value>,

"firmwarestate":<String_value>,

"adapter_id":<Integer_value>,

"mec":<String_value>,

"disk_index":<Integer_value>,

"host_ip_address":<String_value>,

"oec":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,
```

```
"is_used":<Boolean_value>,

"enclosure":<Integer_value>,

"inquiry":<String_value>,

"wwn":<String_value>,

"disk_name":<String_value>,

"physicalstate":<String_value>,

"foreignstate":<String_value>,

"mediatype":<String_value>,

"partner_state":<String_value>,

"response":<String_value>,

"force_clean":<Boolean_value>}]}
```

replace_missing

URL: https://<MGMT-IP>/nitro/v2/config/physical_disk?action=replace_missing;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"physical_disk": {

"logical_disk_name":<String_value>,
```

```
"host_ip_address":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,

"force_clean":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"physical_disk":[{

"ld_number":<Integer_value>,

"deviceid":<Integer_value>,

"logical_disk_name":<String_value>,

"size":<String_value>,

"firmwarestate":<String_value>,

"adapter_id":<Integer_value>,

"mec":<String_value>,

"disk_index":<Integer_value>,

"host_ip_address":<String_value>,

"oec":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,
```

```
"is_used":<Boolean_value>,

"enclosure":<Integer_value>,

"inquiry":<String_value>,

"wwn":<String_value>,

"disk_name":<String_value>,

"physicalstate":<String_value>,

"foreignstate":<String_value>,

"mediatype":<String_value>,

"partner_state":<String_value>,

"response":<String_value>,

"force_clean":<Boolean_value>}]}
```

remove_pd

URL: https://<MGMT-IP>/nitro/v2/config/physical_disk?action=remove_pd;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"physical_disk": {

"logical_disk_name":<String_value>,
```

```
"host_ip_address":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,

"force_clean":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"physical_disk":[{

"ld_number":<Integer_value>,

"deviceid":<Integer_value>,

"logical_disk_name":<String_value>,

"size":<String_value>,

"firmwarestate":<String_value>,

"adapter_id":<Integer_value>,

"mec":<String_value>,

"disk_index":<Integer_value>,

"host_ip_address":<String_value>,

"oec":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,
```

```
"is_used":<Boolean_value>,

"enclosure":<Integer_value>,

"inquiry":<String_value>,

"wwn":<String_value>,

"disk_name":<String_value>,

"physicalstate":<String_value>,

"foreignstate":<String_value>,

"mediatype":<String_value>,

"partner_state":<String_value>,

"response":<String_value>,

"force_clean":<Boolean_value>}]}
```

locate_pd_start

URL: https://<MGMT-IP>/nitro/v2/config/physical_disk?action=locate_pd_start;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"physical_disk": {

"logical_disk_name":<String_value>,
```

```
"host_ip_address":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,

"force_clean":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"physical_disk":[{

"ld_number":<Integer_value>,

"deviceid":<Integer_value>,

"logical_disk_name":<String_value>,

"size":<String_value>,

"firmwarestate":<String_value>,

"adapter_id":<Integer_value>,

"mec":<String_value>,

"disk_index":<Integer_value>,

"host_ip_address":<String_value>,

"oec":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,
```

```
"is_used":<Boolean_value>,

"enclosure":<Integer_value>,

"inquiry":<String_value>,

"wwn":<String_value>,

"disk_name":<String_value>,

"physicalstate":<String_value>,

"foreignstate":<String_value>,

"mediatype":<String_value>,

"partner_state":<String_value>,

"response":<String_value>,

"force_clean":<Boolean_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/physical_disk **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/physical_disk **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of physical_disk resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/physical_disk **? pagesize=#no;pageno=#no** Use this query-parameter to get the physical_disk resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/physical_disk **?count=yes** Use this query-parameter to get the count of physical_disk resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"physical_disk":[{

"ld_number":<Integer_value>,

"deviceid":<Integer_value>,

"logical_disk_name":<String_value>,

"size":<String_value>,

"firmwarestate":<String_value>,

"adapter_id":<Integer_value>,

"mec":<String_value>,

"disk_index":<Integer_value>,

"host_ip_address":<String_value>,

"oec":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,
```

```
"is_used":<Boolean_value>,

"enclosure":<Integer_value>,

"inquiry":<String_value>,

"wwn":<String_value>,

"disk_name":<String_value>,

"physicalstate":<String_value>,

"foreignstate":<String_value>,

"mediatype":<String_value>,

"partner_state":<String_value>,

"response":<String_value>,

"force_clean":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/physical_disk/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"physical_disk":[{

"ld_number":<Integer_value>,

"deviceid":<Integer_value>,

"logical_disk_name":<String_value>,

"size":<String_value>,

"firmwarestate":<String_value>,

"adapter_id":<Integer_value>,

"mec":<String_value>,

"disk_index":<Integer_value>,

"host_ip_address":<String_value>,

"oec":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,

"is_used":<Boolean_value>,

"enclosure":<Integer_value>,

"inquiry":<String_value>,

"wwn":<String_value>,

"disk_name":<String_value>,

"physicalstate":<String_value>,

"foreignstate":<String_value>,
```

```
"mediatype":<String_value>,

"partner_state":<String_value>,

"response":<String_value>,

"force_clean":<Boolean_value>}]}
```

locate_pd_stop

URL: https://<MGMT-IP>/nitro/v2/config/physical_disk?action=locate_pd_stop;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"physical_disk": {

"logical_disk_name":<String_value>,

"host_ip_address":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,

"force_clean":<Boolean_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"physical_disk":[{

"ld_number":<Integer_value>,

"deviceid":<Integer_value>,

"logical_disk_name":<String_value>,

"size":<String_value>,

"firmwarestate":<String_value>,

"adapter_id":<Integer_value>,

"mec":<String_value>,

"disk_index":<Integer_value>,

"host_ip_address":<String_value>,

"oec":<String_value>,

"id":<String_value>,

"slot":<Integer_value>,

"is_used":<Boolean_value>,

"enclosure":<Integer_value>,

"inquiry":<String_value>,

"wwn":<String_value>,

"disk_name":<String_value>,

"physicalstate":<String_value>,

"foreignstate":<String_value>,
```

```
"mediatype":<String_value>,

"partner_state":<String_value>,

"response":<String_value>,

"force_clean":<Boolean_value>}]}
```

snmp_interface

Configuration for XenServer Interface resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
inOctets	<String>	Read-write	The total number of octets received on the interface.
outOctets	<String>	Read-write	The total number of octets transmitted out of the interface.
outNUcastPkts	<String>	Read-write	Number of braoadcast or multicast packets transmitted from this interface.
inDiscards	<String>	Read-write	Number of discarded inbound packets.
speed	<String>	Read-write	Interface's current bandwidth in bits per second.
mtu	<Integer>	Read-write	Range for Base Mac Address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
outDiscards	<String>	Read-write	Number of discarded outbound packets.
outErrors	<String>	Read-write	Number of outbound packets that could not be transmitted because of errors.
descr	<String>	Read-write	Interface Description. Minimum length = 1 Maximum length = 50
inUnknownProtos	<String>	Read-write	Number of packets recieved with unknown or unsupportedprotocol.
index	<Integer>	Read-write	SNMP Index for interfaces and Channels. Minimum value = 1
lastChange	<Integer>	Read-write	Time when the interface entered its current operational state.
outUcastPkts	<String>	Read-write	Number of unicast packets transmitted from this interface.
inNUcastPkts	<String>	Read-write	Number of Broadcast and Multicast packets recieved on this interface.
outQLen	<String>	Read-write	The length of the output packet queue (in packets).

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
type	<Integer>	Read-write	The type of interface.
inUcastPkts	<String>	Read-write	Number of Unicast packets recieved on this interface.
inErrors	<String>	Read-write	Number of inbound packets that contained errors.
operStatus	<Integer>	Read-only	The current operational state of the interface: up(1), down(2), testing(3), unknown(4), dormant(5), notPresent(6) or lowerLayerDown(7).
adminStatus	<Integer>	Read-only	State of the Interface eg. up, down or testing.
ifSpecific	<String>	Read-only	A reference to MIB definitions specific to the particular media being used to realize the interface.
physAddress	<String>	Read-only	Physical Address of the interface.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

virtual_nmi

Configuration for Virtual NMI generate resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
sync_operation	<Boolean>	Read-write	sync operation.
host_ip_address	<String>	Read-write	Host IP Address of the VM on which we

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			are sending virtual NMI signal.
type	<String>	Read-write	Type of device. Minimum length = 1 Maximum length = 64
ip_address	<String>	Read-write	IP Address of the VM on which we are sending virtual NMI signal.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- INITIATE_CRASH_DUMP

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

initiate_crash_dump

URL: https://<MGMT-IP>/nitro/v2/config/virtual_nmi?action=initiate_crash_dump;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"virtual_nmi": {

"host_ip_address":<String_value>,

"ip_address":<String_value>,

"sync_operation":<Boolean_value>,

"type":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"virtual_nmi":[{

"sync_operation":<Boolean_value>,

"act_id":<String_value>,

"host_ip_address":<String_value>,

"type":<String_value>,

"ip_address":<String_value>}]}
```

xen

Configuration for Xen resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hostname	<String>	Read-write	Assign hostname to managed device, if this is not provided, name will be set as host name . Minimum length = 1 Maximum length = 256
std_bw_config	<Integer>	Read-write	Standard Bandwidth running.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
gateway_ipv6	<String>	Read-write	Gateway IPv6 Address.
name	<String>	Read-write	Name of managed device. Minimum length = 1 Maximum length = 128
description	<String>	Read-write	Description of managed device. Minimum length = 1 Maximum length = 512
ent_bw_available	<Integer>	Read-write	Enterprise Bandwidth configured.
mastools_version	<String>	Read-write	Mastools version if the device is embedded agent.
xen_mgmt_ip_address	<String>	Read-write	IP Address for xen server.
ent_bw_total	<Integer>	Read-write	Enterprise Bandwidth Total.
vcpu_config	<Integer>	Read-write	Number of vCPU allocated for the device.
netmask	<String>	Read-write	Netmask of managed device. Minimum length = 1 Maximum length = 64

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ent_bw_config	<Integer>	Read-write	Enterprise Bandwidth configured.
datacenter_id	<String>	Read-write	Datacenter Id is system generated key for data center.
is_managed	<Boolean>	Read-write	Is Managed.
instance_mode	<String>	Read-write	Denotes state-primary,secondary,clip,clusternode.
nexthop_v6	<String>	Read-write	Next Hop IPv6 Address.
ipv4_address	<String>	Read-write	IPv4 Address. Minimum length = 1 Maximum length = 64
xen_mgmt_dns	<String>	Read-write	IP Address for this managed device.
profile_name	<String>	Read-write	Device Profile Name that is attached with this managed device. Minimum length = 1 Maximum length = 128
std_bw_available	<Integer>	Read-write	Standard Bandwidth Available.
internal_ip_address	<String>	Read-write	Internal IP Address for this managed device.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
xen_mgmt_netmask	<String>	Read-write	IP Address for this managed device.
ifCount	<Integer>	Read-write	Number of interfaces added in device.
last_updated_time	<Double>	Read-write	Last Updated Time.
xen_mgmt_gateway	<String>	Read-write	IP Address for this managed device.
plt_bw_total	<Integer>	Read-write	Total Platinum Bandwidth.
id	<String>	Read-write	Id is system generated key for all the managed devices.
ipv6_address	<String>	Read-write	IPv6 Address.
mgmt_ip_address	<String>	Read-write	Management IP Address for this Managed Device. Minimum length = 1 Maximum length = 64
license_edition	<String>	Read-write	Edition of instance.
plt_bw_available	<Integer>	Read-write	Platinum Bandwidth Available.
device_family	<String>	Read-write	Device Family. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
license_grace_time	<Integer>	Read-write	Grace for this Citrix Instance..
type	<String>	Read-write	Type of device, (Xen | NS). Minimum length = 1 Maximum length = 64
gateway	<String>	Read-write	Default Gateway of managed device. Minimum length = 1 Maximum length = 64
config_type	<Integer>	Read-write	Configuration Type. Values: 0: IPv4, 1: IPv6, 2: Both. Maximum value =
if_internal_ip_enabled	<Boolean>	Read-write	Set as true if VPX is managed by internal network (not required to be set for SDWAN).
node_id	<String>	Read-write	Node identification of a device.
xen_mgmt_ifc	<String>	Read-write	IP Address for this managed device.
ip_address	<String>	Read-write	IP Address for this managed device. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64
display_name	<String>	Read-write	Display Name for this managed device. For HA pair it will be A-B, and for Cluster it will be CLIP. Minimum length = 1 Maximum length = 128
std_bw_total	<Integer>	Read-write	Standard Bandwidth.
nexthop	<String>	Read-write	Next Hop IP address. Minimum length = 1 Maximum length = 64
plt_bw_config	<Integer>	Read-write	Platinum Bandwidth configured.
profile_username	<String>	Read-write	User Name specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128
profile_password	<String>	Read-write	Password specified by the user for this Citrix ADC Instance.. Minimum length = 1 Maximum length = 128

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ns_ip_arr_to_save	<String[]>	Read-write	List of IP address for which config needs to be saved.
bios_version	<String>	Read-only	BIOS Version.
acu_total	<Integer>	Read-only	Total Asymmetric Crypto Units available in host.
instance_state	<String>	Read-only	State of device, UP only if device accessible.
enabled	<Boolean>	Read-only	host State.
productcode	<String>	Read-only	XenServer Product Code.
reason	<String>	Read-only	Reason of failure for this managed device.
disk_space	<Double>	Read-only	Show Disk Space (MB) available on host.
disk_free	<Double>	Read-only	Show free Disk Space (MB) available on host.
upsince	<String>	Read-only	Upsince of managed device.
disk_usage	<Double>	Read-only	Disk usage (MB) for host.
tenant_id	<String>	Read-only	Tenant ID.
edition	<String>	Read-only	XenServer Edition.
device_uuid	<String>	Read-only	Device UUID.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
platform_version	<String>	Read-only	SDX platform version.
do_config	<Boolean>	Read-only	Do default config for managed device.
tx	<Double>	Read-only	Out Throughput (Mbps) of host.
version	<String>	Read-only	Device Version.
host_id	<String>	Read-only	Host ID.
number_of_cpu	<Integer>	Read-only	Number of total CPU of host.
memory_free	<Double>	Read-only	Free memory available (MB) in host.
build_number	<String>	Read-only	XenServer Build Number.
sysid	<String>	Read-only	System ID.
uuid	<String>	Read-only	UUID of host.
scu_total	<Integer>	Read-only	Total Symmetric Crypto Units available in host.
uptime	<String>	Read-only	Uptime of device.
cpu_license_type	<Integer>	Read-only	VCPU license 0 = No VCPU License, 1 = VCPU Pool license.
expiry	<String>	Read-only	XenServer Expiry.
supplemental_pack	<String>	Read-only	Supplemental Pack installed on XenServer.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
is_xen_security_supported	<Boolean>	Read-only	Flag indicating whether Xen security can be enabled.
iscsi_iqn	<String>	Read-only	iSCSI IQN.
scu_free	<Integer>	Read-only	Free Symmetric Crypto Units.
total_crypto_devices	<Integer>	Read-only	Total Crypto Devices.
acu_free	<Integer>	Read-only	Free Asymmetric Crypto Units.
cpu_usage	<Double>	Read-only	CPU Usage (%) of host.
is_xen_reachable	<Boolean>	Read-only	Flag indicating whether Xen is reachable.
memory_total	<Double>	Read-only	Total memory of host in MB.
status	<String>	Read-only	Status of managed device.
build_date	<String>	Read-only	XenServer Build Date.
systemname	<String>	Read-only	Device System Name.
version_long	<String>	Read-only	XenServer Version (full text).
version_short	<String>	Read-only	XenServer Version.
kernel_version	<String>	Read-only	Kernel Version.
rx	<Double>	Read-only	In Throughput (Mbps) of host.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
ssl_cores_total	<Integer>	Read-only	Total SSL Cores available in host.
memory_usage	<Double>	Read-only	Memory Usage (%) of host.
number_of_cpu_cores	<Integer>	Read-only	Number of total CPU cores of host.
ssl_cores_free	<Integer>	Read-only	Free SSL Cores.
act_id	<String>	Read-only	Activity Id.

Operations

(click to see Properties)

- REBOOT
- STOP
- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

reboot

URL: https://<MGMT-IP>/nitro/v2/config/xen/id_value<String>?action=reboot;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"xen": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

stop

URL: https://<MGMT-IP>/nitro/v2/config/xen/id_value<String>?action=stop;onerror=<String_value>
HTTPS Method: POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"xen": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen **?count=yes** Use this query-parameter to get the count of xen resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen":[{

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"bios_version":<String_value>,

"std_bw_config":<Integer_value>,

"acu_total":<Integer_value>,

"gateway_ipv6":<String_value>,

"instance_state":<String_value>,

"enabled":<Boolean_value>,
```

```
"productcode":<String_value>,

"name":<String_value>,

"reason":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"disk_free":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"user_driven":<Boolean_value>,

"mastools_version":<String_value>,

"xen_mgmt_ip_address":<String_value>,

"disk_usage":<Double_value>,

"ent_bw_total":<Integer_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"prev_xen_rx":<Double_value>,

"edition":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"platform_version":<String_value>,

"ent_bw_config":<Integer_value>,
```

```
"do_config":<Boolean_value>,

"tx":<Double_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"xen_rx":<Double_value>,

"is_managed":<Boolean_value>,

"instance_mode":<String_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,

"memory_free":<Double_value>,

"build_number":<String_value>,

"ipv4_address":<String_value>,

"xen_mgmt_dns":<String_value>,

"prev_xen_tx":<Double_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"internal_ip_address":<String_value>,

"xen_mgmt_netmask":<String_value>,

"sysid":<String_value>,

"uuid":<String_value>,
```

```
"ifCount":<Integer_value>,

"scu_total":<Integer_value>,

"last_updated_time":<Double_value>,

"xen_mgmt_gateway":<String_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,

"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"expiry":<String_value>,

"device_family":<String_value>,

"supplemental_pack":<String_value>,

"license_grace_time":<Integer_value>,

"is_xen_security_supported":<Boolean_value>,

"iscsi_iqn":<String_value>,

"scu_free":<Integer_value>,

"xen_tx":<Double_value>,

"total_crypto_devices":<Integer_value>,

"acu_free":<Integer_value>,
```

```
"cpu_usage":<Double_value>,

"type":<String_value>,

"is_xen_reachable":<Boolean_value>,

"memory_total":<Double_value>,

"status":<String_value>,

"gateway":<String_value>,

"build_date":<String_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,

"node_id":<String_value>,

"version_long":<String_value>,

"xen_mgmt_ifc":<String_value>,

"ip_address":<String_value>,

"version_short":<String_value>,

"kernel_version":<String_value>,

"prev_poll_time":<Double_value>,

"ping_state":<Integer_value>,

"rx":<Double_value>,

"ssl_cores_total":<Integer_value>,

"memory_usage":<Double_value>,

"display_name":<String_value>,
```

```
"std_bw_total":<Integer_value>,

"number_of_cpu_cores":<Integer_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"parent_inventory":<Boolean_value>,

"profile_username":<String_value>,

"profile_password":<String_value>,

"default_password":<String_value>,

"ssl_cores_free":<Integer_value>,

"ns_ip_arr_to_save":<String_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen":[{

"is_grace":<Boolean_value>,

"hostname":<String_value>,

"bios_version":<String_value>,

"std_bw_config":<Integer_value>,

"acu_total":<Integer_value>,

"gateway_ipv6":<String_value>,

"instance_state":<String_value>,

"enabled":<Boolean_value>,

"productcode":<String_value>,

"name":<String_value>,

"reason":<String_value>,

"disk_space":<Double_value>,

"description":<String_value>,

"ent_bw_available":<Integer_value>,

"disk_free":<Double_value>,

"is_pooled_license":<Boolean_value>,

"upsince":<String_value>,

"user_driven":<Boolean_value>,
```

```
"mastools_version":<String_value>,

"xen_mgmt_ip_address":<String_value>,

"disk_usage":<Double_value>,

"ent_bw_total":<Integer_value>,

"tenant_id":<String_value>,

"vcpu_config":<Integer_value>,

"prev_xen_rx":<Double_value>,

"edition":<String_value>,

"netmask":<String_value>,

"device_uuid":<String_value>,

"platform_version":<String_value>,

"ent_bw_config":<Integer_value>,

"do_config":<Boolean_value>,

"tx":<Double_value>,

"version":<String_value>,

"host_id":<String_value>,

"datacenter_id":<String_value>,

"xen_rx":<Double_value>,

"is_managed":<Boolean_value>,

"instance_mode":<String_value>,

"number_of_cpu":<Integer_value>,

"nexthop_v6":<String_value>,
```

```
"memory_free":<Double_value>,

"build_number":<String_value>,

"ipv4_address":<String_value>,

"xen_mgmt_dns":<String_value>,

"prev_xen_tx":<Double_value>,

"profile_name":<String_value>,

"std_bw_available":<Integer_value>,

"seq_no":<Double_value>,

"internal_ip_address":<String_value>,

"xen_mgmt_netmask":<String_value>,

"sysid":<String_value>,

"uuid":<String_value>,

"ifCount":<Integer_value>,

"scu_total":<Integer_value>,

"last_updated_time":<Double_value>,

"xen_mgmt_gateway":<String_value>,

"uptime":<String_value>,

"plt_bw_total":<Integer_value>,

"id":<String_value>,

"ipv6_address":<String_value>,

"mgmt_ip_address":<String_value>,

"license_edition":<String_value>,
```

```
"plt_bw_available":<Integer_value>,

"cpu_license_type":<Integer_value>,

"expiry":<String_value>,

"device_family":<String_value>,

"supplemental_pack":<String_value>,

"license_grace_time":<Integer_value>,

"is_xen_security_supported":<Boolean_value>,

"iscsi_iqn":<String_value>,

"scu_free":<Integer_value>,

"xen_tx":<Double_value>,

"total_crypto_devices":<Integer_value>,

"acu_free":<Integer_value>,

"cpu_usage":<Double_value>,

"type":<String_value>,

"is_xen_reachable":<Boolean_value>,

"memory_total":<Double_value>,

"status":<String_value>,

"gateway":<String_value>,

"build_date":<String_value>,

"config_type":<Integer_value>,

"systemname":<String_value>,

"if_internal_ip_enabled":<Boolean_value>,
```

```
"node_id":<String_value>,

"version_long":<String_value>,

"xen_mgmt_ifc":<String_value>,

"ip_address":<String_value>,

"version_short":<String_value>,

"kernel_version":<String_value>,

"prev_poll_time":<Double_value>,

"ping_state":<Integer_value>,

"rx":<Double_value>,

"ssl_cores_total":<Integer_value>,

"memory_usage":<Double_value>,

"display_name":<String_value>,

"std_bw_total":<Integer_value>,

"number_of_cpu_cores":<Integer_value>,

"nexthop":<String_value>,

"plt_bw_config":<Integer_value>,

"parent_inventory":<Boolean_value>,

"profile_username":<String_value>,

"profile_password":<String_value>,

"default_password":<String_value>,

"ssl_cores_free":<Integer_value>,

"ns_ip_arr_to_save":<String_value>,
```

```
"act_id":<String_value>,

"sync_operation":<Boolean_value>}]}
```

xen_applied_hotfix

Configuration for Xen Applied Hotfixes resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-only	Hotfix Name.
description	<String>	Read-only	Description of the Hotfix.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_applied_hotfix **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_applied_hotfix **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_applied_hotfix resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_applied_hotfix **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_applied_hotfix resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_applied_hotfix **?count=yes** Use this query-parameter to get the count of xen_applied_hotfix resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_applied_hotfix":[{

"name":<String_value>,

"description":<String_value>}]}
```

xen_bluecatvpx_image

Configuration for XVA images of bluecat Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/xen_bluecatvpx_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_bluecatvpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_bluecatvpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_bluecatvpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_bluecatvpx_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_bluecatvpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_bluecatvpx_image **?count=yes** Use this query-parameter to get the count of xen_bluecatvpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_bluecatvpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_bluecatvpx_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_bluecatvpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/xen_bluecatvpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

xen_brvpx_image

Configuration for XVA images of Repeater Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 256
file_size	<Double>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/xen_brvpx_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_brvpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_brvpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of

xen_brvpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_brvpx_image **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_brvpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_brvpx_image **?count=yes** Use this query-parameter to get the count of xen_brvpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_brvpx_image":[{

"file_location_path":<String_value>,

"file_size":<Double_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_brvpx_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_brvpx_image":[{

"file_location_path":<String_value>,

"file_size":<Double_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/xen_brvpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

xen_cb_host_interface

Configuration for Assign Host Interface to CB resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
skip_reboot	<Boolean>	Read-write	Perform an instance reboot when assigning interface.
vm_ip	<String>	Read-write	IP Address of VM to assign Host Interface. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 64

Operations

(click to see Properties)

- ADD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/xen_cb_host_interface?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"xen_cb_host_interface": {

"skip_reboot":<Boolean_value>,

"vm_ip":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_cb_host_interface":[{

"skip_reboot":<Boolean_value>,

"vm_ip":<String_value>}]}
```

xen_core

Configuration for shows SDX core details resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hyper_threads	<xen_hyper_thread[]>	Read-write	Hyper threads belonging to this core.
usage	<Double>	Read-only	Hyperthread Usage (%).
allocation_mode	<String>	Read-only	Available, Commited, SharedDedicated, Pooled, SemiPooled, StrictReserved, LooseReserved.
socket_id	<Integer>	Read-only	CPU Socket ID to which this core belongs.
core_id	<Integer>	Read-only	Core .
vm_names	<String>	Read-only	VM Names.
admin_domain_name	<String>	Read-only	Admin Domain to which this core belongs.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

xen_cpu_layout

Configuration for shows complete layout of the CPU resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
cores	<xen_core[]>	Read-write	array of cores.
is_single_pool	<Boolean>	Read-only	Pool Allocation Mode.
num_sockets	<Integer>	Read-only	Number Of Sockets.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
available_cpu_capacity_in_dedicated_mode	<Integer>	Read-only	Capacity In Terms Of Dedicated VMs.
available_cpu_capacity_in_shared_mode	<Integer>	Read-only	Capacity In Terms Of Shared VMs.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_cpu_layout **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_cpu_layout **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_cpu_layout resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_cpu_layout **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_cpu_layout resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_cpu_layout **?count=yes** Use this query-parameter to get the count of xen_cpu_layout resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_cpu_layout":[{

"is_single_pool":<Boolean_value>,

"num_sockets":<Integer_value>,

"available_cpu_capacity_in_dedicated_mode":<Integer_value>,

"available_cpu_capacity_in_shared_mode":<Integer_value>,

"cores":[{

"usage":<Double_value>,

"allocation_mode":<String_value>,

"socket_id":<Integer_value>,

"core_id":<Integer_value>,

"vm_names":<String_value>,

"admin_domain_name":<String_value>,

"hyper_threads":[{

"usage":<Double_value>,

"thread_id":<Integer_value>,

"shared_vm_list":<String_value>,

"dedicated_vm_name":<String_value>,

"used_for":<String_value>,

"allocation_mode":<String_value>}]}]}]}
```

xen_cpu_switches

Configuration for various swicthes used for CPU allocation to the VMs resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
id	<String>	Read-write	Id is system generated key for all the CPUs.
use_new_schema_for_decapolis	<Boolean>	Read-only	Total Thread Count.
use_both_sockets_for_smaller_vms	<Boolean>	Read-only	Total Thread Count.
shared_pool_across_sockets	<Boolean>	Read-only	Total Thread Count.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_cpu_switches **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_cpu_switches **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_cpu_switches resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_cpu_switches **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_cpu_switches resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_cpu_switches **?count=yes** Use this query-parameter to get the count of xen_cpu_switches resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_cpu_switches":[{

"use_new_schema_for_decapolis":<Boolean_value>,

"use_both_sockets_for_smaller_vms":<Boolean_value>,

"id":<String_value>,

"shared_pool_across_sockets":<Boolean_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_cpu_switches/id_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_cpu_switches":[{

"use_new_schema_for_decapolis":<Boolean_value>,

"use_both_sockets_for_smaller_vms":<Boolean_value>,

"id":<String_value>,

"shared_pool_across_sockets":<Boolean_value>}]}
```

xen_health_disk

Configuration for Disk IO Statistics resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
total_blocks_read	<Double>	Read-write	Total blocks read.
status	<String>	Read-write	Status of the storage repository - OK/ ERROR/ DEGRADED.
name	<String>	Read-write	Name of the Disk.
sr_name	<String>	Read-write	Name of the storage repository.
size	<Double>	Read-write	Size of the disk (bytes).
total_blocks_written	<Double>	Read-write	Total blocks written.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
physical_disks	<String>	Read-write	The physical disks that make up the logical disk.
block_read_rate	<Double>	Read-write	Blocks read per second.
block_write_rate	<Double>	Read-write	Blocks written per second.
bay_number	<String>	Read-write	Bay number.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
utilized	<Double>	Read-write	Utilization of the disk (bytes).
transaction_rate	<Double>	Read-write	Transactions per second.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_disk **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_disk **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_disk resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_disk **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_disk resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_disk **?count=yes** Use this query-parameter to get the count of xen_health_disk resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_disk":[{

"total_blocks_read":<Double_value>,

"status":<String_value>,

"name":<String_value>,

"sr_name":<String_value>,

"size":<Double_value>,

"total_blocks_written":<Double_value>,

"physical_disks":<String_value>,

"block_read_rate":<Double_value>,

"block_write_rate":<Double_value>,

"bay_number":<String_value>,

"host_ip_address":<String_value>,

"utilized":<Double_value>,
```

```
"transaction_rate":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_disk/bay_number_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_disk":[{

"total_blocks_read":<Double_value>,

"status":<String_value>,

"name":<String_value>,

"sr_name":<String_value>,

"size":<Double_value>,

"total_blocks_written":<Double_value>,
```

```
"physical_disks":<String_value>,

"block_read_rate":<Double_value>,

"block_write_rate":<Double_value>,

"bay_number":<String_value>,

"host_ip_address":<String_value>,

"utilized":<Double_value>,

"transaction_rate":<Double_value>}]}
```

xen_health_ftw

Configuration for FTW Health resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
pair_port	<String>	Read-write	Pair Name. Minimum length = 1 Maximum length = 10
mode	<String>	Read-write	Mode. Minimum length = 1 Maximum length = 32
mapped_port	<String>	Read-write	Mapped Port Name Ex: eth0. Minimum length = 1

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 10
display_name	<String>	Read-write	Display Name Ex: Management Port. Minimum length = 1 Maximum length = 50
port	<String>	Read-write	Port Name Ex: 10/1. Minimum length = 1 Maximum length = 10
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
pair_display_name	<String>	Read-write	Pair Display Name Ex: Management Port. Minimum length = 1 Maximum length = 50
state	<String>	Read-only	State of the Interface - UP/DOWN.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_ftw **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_ftw **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_ftw resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_ftw **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_ftw resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_ftw **?count=yes** Use this query-parameter to get the count of xen_health_ftw resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_ftw":[{

"pair_port":<String_value>,

"mode":<String_value>,

"mapped_port":<String_value>,

"display_name":<String_value>,

"port":<String_value>,

"state":<String_value>,
```

```
"host_ip_address":<String_value>,

"pair_display_name":<String_value>,

"ha_enabled":<String_value>,

"version":<String_value>,

"status":<String_value>,

"l2_enabled":<String_value>,

"message":<String_value>,

"error_code":<Integer_value>}]}
```

xen_health_hsm

Configuration for HSM Statistics resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
partition_name	<String>	Read-write	FIPS Partition Name.
rsa_operations	<Integer>	Read-only	Number of RSA operations in flight.
crypto_utilization	<Double>	Read-only	Utilization of the crypto resources allocated for each partition as a percentage of total allocated..
rsa_handshakes	<Integer>	Read-only	Total number of new RSA handshakes processed.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
maximum_key_store_size	<Integer>	Read-only	Maximum no of keys for the partition.
queue_depth	<Integer>	Read-only	Queue depth of instruction queues allocated to the instance from each partition.
ssl_bulk_requests	<Integer>	Read-only	Total number of SSL bulk requests in flight.
available_key_store_size	<Integer>	Read-only	Available no of keys for the partition.
ssl_requests_processed	<Integer>	Read-only	Total number of SSL bulk record encryption/ decryption requests processed.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_hsm **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_hsm **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_hsm resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_hsm **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_hsm resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_hsm **?count=yes** Use this query-parameter to get the count of xen_health_hsm resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_hsm":[{

"rsa_operations":<Integer_value>,

"crypto_utilization":<Double_value>,

"rsa_handshakes":<Integer_value>,

"maximum_key_store_size":<Integer_value>,

"queue_depth":<Integer_value>,

"partition_name":<String_value>,

"ssl_bulk_requests":<Integer_value>,

"available_key_store_size":<Integer_value>,

"ssl_requests_processed":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_hsm/partition_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_hsm":[{

"rsa_operations":<Integer_value>,

"crypto_utilization":<Double_value>,

"rsa_handshakes":<Integer_value>,

"maximum_key_store_size":<Integer_value>,

"queue_depth":<Integer_value>,

"partition_name":<String_value>,

"ssl_bulk_requests":<Integer_value>,

"available_key_store_size":<Integer_value>,

"ssl_requests_processed":<Integer_value>}]}
```

xen_health_interface

Configuration for XenServer Interface Health resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
rx_packets	<Double>	Read-write	Received packets.
tx_packets	<Double>	Read-write	Transmitted packets.
mapped_port	<String>	Read-write	Mapped Port Name Ex: eth0. Minimum length = 1 Maximum length = 10
display_name	<String>	Read-write	Display Name Ex: Management Port. Minimum length = 1 Maximum length = 50
port	<String>	Read-write	Port Name Ex: 10/1. Minimum length = 1 Maximum length = 10
rx_errors	<Double>	Read-write	Receiving errors.
rx_bytes	<Double>	Read-write	Received bytes.
tx_bytes	<Double>	Read-write	Transmitted bytes.
tx_errors	<Double>	Read-write	Transmission errors.
host_ip_address	<String>	Read-write	Host IP Address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 64
vf_total	<Integer>	Read-write	Total Virtual Functions.
vf_assigned	<Integer>	Read-write	Assigned Virtual Functions.
state	<String>	Read-only	State of the Interface - UP/DOWN.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_interface **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_interface **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_interface resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_interface **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_interface resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_interface **?count=yes** Use this query-parameter to get the count of xen_health_interface resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_interface":[{

"rx_packets":<Double_value>,

"tx_packets":<Double_value>,

"mapped_port":<String_value>,

"display_name":<String_value>,

"port":<String_value>,

"state":<String_value>,

"rx_errors":<Double_value>,

"rx_bytes":<Double_value>,

"tx_bytes":<Double_value>,

"tx_errors":<Double_value>,

"host_ip_address":<String_value>,

"vf_total":<Integer_value>,

"vf_assigned":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_interface/mapped_port_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_interface":[{

"rx_packets":<Double_value>,

"tx_packets":<Double_value>,

"mapped_port":<String_value>,

"display_name":<String_value>,

"port":<String_value>,

"state":<String_value>,

"rx_errors":<Double_value>,

"rx_bytes":<Double_value>,

"tx_bytes":<Double_value>,
```

```
"tx_errors":<Double_value>,

"host_ip_address":<String_value>,

"vf_total":<Integer_value>,

"vf_assigned":<Integer_value>}]}
```

xen_health_monitor

Configuration for XenServer Health Monitor (IPMI Sensor list) resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
unit	<String>	Read-write	Measurement Unit.
current_value	<String>	Read-write	Current Value of the IPMI Sensor.
status	<String>	Read-write	IPMI Sensor Status - OK/ERROR/NA.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
name	<String>	Read-write	IPMI Sensor Name.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_monitor **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_monitor **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_monitor resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_monitor **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_monitor resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_monitor **?count=yes** Use this query-parameter to get the count of xen_health_monitor resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_monitor":[{

"unit":<String_value>,

"current_value":<String_value>,

"number_of_failures":<Integer_value>,

"status":<String_value>,

"host_ip_address":<String_value>,

"name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_monitor/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_monitor":[{

"unit":<String_value>,

"current_value":<String_value>,

"number_of_failures":<Integer_value>,

"status":<String_value>,

"host_ip_address":<String_value>,

"name":<String_value>}]}
```

xen_health_monitor_fan_speed

Configuration for Xen Health Monitor (IPMI Sensor) - Fan Speed resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	IPMI Sensor Name.
unit	<String>	Read-write	Measurement Unit.
current_value	<String>	Read-write	Current Value of the IPMI Sensor.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
status	<String>	Read-write	IPMI Sensor Status - OK/ERROR/NA.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_fan_speed **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_fan_speed **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_monitor_fan_speed resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_fan_speed **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_monitor_fan_speed resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_fan_speed **?count=yes** Use this query-parameter to get the count of xen_health_monitor_fan_speed resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_monitor_fan_speed":[{

"name":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"host_ip_address":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_fan_speed/name_value<String>
HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_monitor_fan_speed":[{

"name":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"host_ip_address":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

xen_health_monitor_misc

Configuration for Xen Health Monitor (IPMI Sensor) - Miscellaneous resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	IPMI Sensor Name.
unit	<String>	Read-write	Measurement Unit.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
current_value	<String>	Read-write	Current Value of the IPMI Sensor.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
status	<String>	Read-write	IPMI Sensor Status - OK/ERROR/NA.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_misc **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_misc **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_monitor_misc resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_misc **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_monitor_misc resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_misc **?count=yes** Use this query-parameter to get the count of xen_health_monitor_misc resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_monitor_misc":[{

"name":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"host_ip_address":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_misc/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_monitor_misc":[{

"name":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"host_ip_address":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

xen_health_monitor_temp

Configuration for Xen Health Monitor (IPMI Sensor) - Temperature resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	IPMI Sensor Name.
unit	<String>	Read-write	Measurement Unit.
current_value	<String>	Read-write	Current Value of the IPMI Sensor.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
status	<String>	Read-write	IPMI Sensor Status - OK/ERROR/NA.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_temp **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_temp **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_monitor_temp resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_temp **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_monitor_temp resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_temp **?count=yes** Use this query-parameter to get the count of xen_health_monitor_temp resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_monitor_temp":[{

"name":<String_value>,

"unit":<String_value>,
```

```
"current_value":<String_value>,

"host_ip_address":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_temp/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_monitor_temp":[{

"name":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,
```

```
"host_ip_address":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

xen_health_monitor_voltage

Configuration for Xen Health Monitor (IPMI Sensor) - Voltage resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	IPMI Sensor Name.
unit	<String>	Read-write	Measurement Unit.
current_value	<String>	Read-write	Current Value of the IPMI Sensor.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
status	<String>	Read-write	IPMI Sensor Status - OK/ERROR/NA.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_voltage **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_voltage **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_monitor_voltage resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_voltage **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_monitor_voltage resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_voltage **?count=yes** Use this query-parameter to get the count of xen_health_monitor_voltage resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_monitor_voltage":[{

"name":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"host_ip_address":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_monitor_voltage/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_monitor_voltage":[{

"name":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"host_ip_address":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

xen_health_resource

Configuration for Xen Health Resource resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
expected_value	<String>	Read-write	Expected value of the resource.
unit	<String>	Read-write	Measurement unit for the resource.
current_value	<String>	Read-write	Current value of the resource.
status	<String>	Read-write	Health status of the resource - OK/ ERROR.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
name	<String>	Read-write	Resource name.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_resource **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_resource **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_resource resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_resource **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_resource resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_resource **?count=yes** Use this query-parameter to get the count of xen_health_resource resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_resource":[{

"expected_value":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"number_of_failures":<Integer_value>,

"status":<String_value>,

"host_ip_address":<String_value>,

"name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_resource/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_resource":[{

"expected_value":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"number_of_failures":<Integer_value>,

"status":<String_value>,

"host_ip_address":<String_value>,

"name":<String_value>}]}
```

xen_health_resource_hw

Configuration for Hardware Resource resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Resource name.
unit	<String>	Read-write	Measurement unit for the resource.
current_value	<String>	Read-write	Current value of the resource.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
expected_value	<String>	Read-write	Expected value of the resource.
status	<String>	Read-write	Health status of the resource - OK/ ERROR.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_resource_hw **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_resource_hw **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_resource_hw resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_resource_hw **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_resource_hw resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_resource_hw **?count=yes** Use this query-parameter to get the count of xen_health_resource_hw resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_resource_hw":[{

"name":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"host_ip_address":<String_value>,

"expected_value":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_resource_hw/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_resource_hw":[{

"name":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"host_ip_address":<String_value>,

"expected_value":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

xen_health_resource_sw

Configuration for Software Resource resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
name	<String>	Read-write	Resource name.
unit	<String>	Read-write	Measurement unit for the resource.
current_value	<String>	Read-write	Current value of the resource.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
expected_value	<String>	Read-write	Expected value of the resource.
status	<String>	Read-write	Health status of the resource - OK/ ERROR.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_resource_sw **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_resource_sw **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_resource_sw resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_resource_sw **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_resource_sw resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_resource_sw **?count=yes** Use this query-parameter to get the count of xen_health_resource_sw resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_resource_sw":[{

"name":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"host_ip_address":<String_value>,

"expected_value":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_resource_sw/name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_resource_sw":[{

"name":<String_value>,

"unit":<String_value>,

"current_value":<String_value>,

"host_ip_address":<String_value>,

"expected_value":<String_value>,

"status":<String_value>,

"number_of_failures":<Integer_value>}]}
```

xen_health_sr

Configuration for Storage Repository resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
status	<String>	Read-write	Status of the storage repository - OK/ ERROR.
name	<String>	Read-write	Name of the storage repository.
size	<Double>	Read-write	Physical size of the storage repository.
bay_number	<String>	Read-write	Bay number.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
utilized	<Double>	Read-write	Physical utilization of the storage repository.
act_id	<String>	Read-only	Activity Id.
is_mirrored	<String>	Read-only	Mirroring Status.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_sr **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_sr **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_sr resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_sr **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_sr resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_sr **?count=yes** Use this query-parameter to get the count of xen_health_sr resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_sr":[{

"status":<String_value>,

"name":<String_value>,

"size":<Double_value>,

"act_id":<String_value>,

"bay_number":<String_value>,

"number_of_failures":<Integer_value>,
```

```
"host_ip_address":<String_value>,

"is_mirrored":<String_value>,

"utilized":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_sr/bay_number_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_sr":[{

"status":<String_value>,

"name":<String_value>,

"size":<Double_value>,

"act_id":<String_value>,
```

```
"bay_number":<String_value>,

"number_of_failures":<Integer_value>,

"host_ip_address":<String_value>,

"is_mirrored":<String_value>,

"utilized":<Double_value>}]}
```

xen_health_ssl_core

Configuration for XenServer SSL Core Health resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
vf_total	<Integer>	Read-write	Total SSL Virtual Functions.
vf_assigned	<Integer>	Read-write	Assigned SSL Virtual Functions.
cpu_socket	<String>	Read-write	CPU Socket for the SSL Core.

Operations

(click to see Properties)

- GET (ALL)
- GET

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_ssl_core **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_ssl_core **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_ssl_core resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_ssl_core **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_ssl_core resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_ssl_core **?count=yes** Use this query-parameter to get the count of xen_health_ssl_core resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_ssl_core":[{

"host_ip_address":<String_value>,

"vf_total":<Integer_value>,

"vf_assigned":<Integer_value>,

"cpu_socket":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_ssl_core/cpu_socket_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_ssl_core":[{

"host_ip_address":<String_value>,

"vf_total":<Integer_value>,

"vf_assigned":<Integer_value>,

"cpu_socket":<String_value>}]}
```

xen_health_summary

Configuration for Health Summary resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
last_poll_time_resources	<String>	Read-write	Last poll time for resources.
last_poll_time_sensors	<String>	Read-write	Last poll time for sensors.
host_ip_address	<String>	Read-write	Host IP Address. Minimum length = 1 Maximum length = 64
last_poll_time_interfaces	<String>	Read-write	Last poll time for interfaces.
id	<String>	Read-write	Id is system generated key.
last_poll_time_storage	<String>	Read-write	Last poll time for storage.
sensor_temp	<String>	Read-write	Temperature sensors summary.
storage_disk	<String>	Read-write	Storage disks summary.
sensor_volt	<String>	Read-write	Voltage sensors summary.
resource_sw	<String>	Read-write	Software resources summary.
resource_hw	<String>	Read-write	Hardware resources summary.
sensor_misc	<String>	Read-write	Miscellaneous sensors summary.
storage_repo	<String>	Read-write	Storage repository summary.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
sensor_fan	<String>	Read-write	Fan speed sensros summary.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_health_summary **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_health_summary **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_health_summary resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_health_summary **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_health_summary resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_health_summary **?count=yes** Use this query-parameter to get the count of xen_health_summary resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_health_summary":[{

"last_poll_time_resources":<String_value>,
```

```
"last_poll_time_sensors":<String_value>,

"host_ip_address":<String_value>,

"last_poll_time_interfaces":<String_value>,

"id":<String_value>,

"last_poll_time_storage":<String_value>,

"sensor_temp":<String_value>,

"storage_disk":<String_value>,

"sensor_volt":<String_value>,

"resource_sw":<String_value>,

"resource_hw":<String_value>,

"sensor_misc":<String_value>,

"interface":<String_value>,

"storage_repo":<String_value>,

"sensor_fan":<String_value>}]}
```

xen_hotfix

Configuration for Xen Hotfix resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
hot_fix_applied	<String>	Read-only	Checking whether host fixes applied or not .
file_last_modified	<String>	Read-only	Last Modified.
after_apply_guidance	<String>	Read-only	Steps required after Hotfix installation.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- APPLY
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

apply

URL: https://<MGMT-IP>/nitro/v2/config/xen_hotfix/file_name_value<String>?action=apply;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"xen_hotfix": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/xen_hotfix/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_hotfix **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_hotfix **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of

xen_hotfix resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_hotfix **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_hotfix resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_hotfix **?count=yes** Use this query-parameter to get the count of xen_hotfix resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_hotfix":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"hot_fix_applied":<String_value>,

"file_last_modified":<String_value>,
```

```
"after_apply_guidance":<String_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_hotfix/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_hotfix":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"hot_fix_applied":<String_value>,

"file_last_modified":<String_value>,

"after_apply_guidance":<String_value>,
```

```
"file_last_modified_epoch":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/xen_hotfix **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

xen_hsm

Configuration for HSM Admin Configuration resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
hsm_fw_sign_file	<String>	Read-write	HSM Signature file.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 256
mco_password	<String>	Read-write	Master crypto officer password. Minimum length = 7 Maximum length = 14
hsm_fw_image	<String>	Read-write	HSM Firmware Image file. Minimum length = 1 Maximum length = 256
mco_username	<String>	Read-write	Master crypto officer user name. Minimum length = 1 Maximum length = 32
occupied_contexts	<Double>	Read-only	Occupied no of contexts.
model	<String>	Read-only	Model of adapater.
serial_no	<String>	Read-only	Serial no of adapater.
occupied_keys	<Double>	Read-only	Occupied no of keys.
total_partitions	<Integer>	Read-only	Total no of partitions.
act_id	<String>	Read-only	Activity Id.
host_ip_address	<String>	Read-only	host_ip_address.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
occupied_partitions	<Integer>	Read-only	Occupied no of partitions.
total_contexts	<Double>	Read-only	Total no of contexts.
label_name	<String>	Read-only	Label of adapater.
total_cores	<Double>	Read-only	Total no of cores.
firmware_version	<String>	Read-only	Firmware version of hardware.
hsm_state	<String>	Read-only	HSM state.
total_keys	<Double>	Read-only	Total no of keys.
occupied_cores	<Double>	Read-only	Occupied no of cores.
build_no	<String>	Read-only	Build no of adapater.
part_no	<String>	Read-only	Part no of adapater.

Operations

(click to see Properties)

- ADD
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

add

URL: https://<MGMT-IP>/nitro/v2/config/xen_hsm?onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"xen_hsm": {

"hsm_fw_sign_file":<String_value>,
```

```
"mco_password":<String_value>,

"mco_username":<String_value>,

"hsm_fw_image":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_hsm":[{

"occupied_contexts":<Double_value>,

"hsm_fw_sign_file":<String_value>,

"model":<String_value>,

"serial_no":<String_value>,

"occupied_keys":<Double_value>,

"total_partitions":<Integer_value>,

"act_id":<String_value>,

"mco_password":<String_value>,

"host_ip_address":<String_value>,

"occupied_partitions":<Integer_value>,

"total_contexts":<Double_value>,

"label_name":<String_value>,

"total_cores":<Double_value>,
```

```
"firmware_version":<String_value>,

"hsm_fw_image":<String_value>,

"mco_username":<String_value>,

"sync_operation":<Boolean_value>,

"hsm_state":<String_value>,

"total_keys":<Double_value>,

"occupied_cores":<Double_value>,

"build_no":<String_value>,

"part_no":<String_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_hsm **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_hsm **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_hsm resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_hsm **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_hsm resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_hsm **?count=yes** Use this query-parameter to get the count of xen_hsm resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_hsm":[{

"occupied_contexts":<Double_value>,

"hsm_fw_sign_file":<String_value>,

"model":<String_value>,

"serial_no":<String_value>,

"occupied_keys":<Double_value>,

"total_partitions":<Integer_value>,

"act_id":<String_value>,

"mco_password":<String_value>,

"host_ip_address":<String_value>,

"occupied_partitions":<Integer_value>,

"total_contexts":<Double_value>,

"label_name":<String_value>,

"total_cores":<Double_value>,

"firmware_version":<String_value>,

"hsm_fw_image":<String_value>,
```

```
"mco_username":<String_value>,

"sync_operation":<Boolean_value>,

"hsm_state":<String_value>,

"total_keys":<Double_value>,

"occupied_cores":<Double_value>,

"build_no":<String_value>,

"part_no":<String_value>}]}
```

xen_hsm_fw_image

Configuration for HSM firmware image file resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/xen_hsm_fw_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_hsm_fw_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_hsm_fw_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_hsm_fw_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_hsm_fw_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_hsm_fw_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_hsm_fw_image **?count=yes** Use this query-parameter to get the count of xen_hsm_fw_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_hsm_fw_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_hsm_fw_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_hsm_fw_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/xen_hsm_fw_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

xen_hyper_thread

Configuration for shows hyper thread details resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
usage	<Double>	Read-only	Hyperthread Usage (%).
thread_id	<Integer>	Read-only	Thread ID.
shared_vm_list	<String>	Read-only	Shared VM List.
dedicated_vm_name	<String>	Read-only	Dedicated VM Name.
used_for	<String>	Read-only	None, PEOnly, MgmtOnly, PENotShared_MgmtShared, PEShared_MgmtShared.
allocation_mode	<String>	Read-only	Available, Commited, SharedDedicated, Pooled, SemiPooled,

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			StrictReserved, LooseReserved.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

xen_image

Configuration for XenServer ISO Image File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/xen_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_image **?count=yes** Use this query-parameter to get the count of xen_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_image":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_dir":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_image":[{

"file_location_path":<String_value>,

"file_size":<Integer_value>,

"file_name":<String_value>,

"file_last_modified":<String_value>,

"is_dir":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/xen_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

xen_lacp_member_stats

Configuration for shows runtime lacp stats per member resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
port_id	<String>	Read-write	Interface name: 0/1, 10/1, 1/1. Maximum length = 50
partner_sync	<String>	Read-only	Syncronization.
actor_sys_priority	<Integer>	Read-only	Actor system priority.
actor_collection	<String>	Read-only	Collection.
actor_port_priority	<Integer>	Read-only	Actor Port priority.
actor_rx_status	<String>	Read-only	RX Status for actor.
actor_select_status	<String>	Read-only	Select Status for actor.
actor_default	<String>	Read-only	Defaulted.
actor_sync	<String>	Read-only	Syncronization.
partner_distribution	<String>	Read-only	Distribution.
actor_sys_mac	<String>	Read-only	actor system mac.
actor_port_no	<Integer>	Read-only	Actor port number.
partner_sys_mac	<String>	Read-only	partner system mac.
partner_port_priority	<Integer>	Read-only	partner port priority.
actor_distribution	<String>	Read-only	Distribution.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
partner_sys_priority	<Integer>	Read-only	partner system priority.
partner_key	<Integer>	Read-only	partner key value.
actor_key	<Integer>	Read-only	actor key value.
actor_mux_status	<String>	Read-only	MUX Status for actor.
partner_collection	<String>	Read-only	Collection.
partner_aggregation	<String>	Read-only	Aggregation.
partner_default	<String>	Read-only	Defaulted.
partner_port_no	<Integer>	Read-only	partner port number.
actor_timeout	<String>	Read-only	Actor timeout value.
partner_mode	<String>	Read-only	partner Mode.
actor_aggregation	<String>	Read-only	Aggregation.
actor_mode	<String>	Read-only	Actor Mode.
port_status	<String>	Read-only	Port Status : UP/ DOWN.
partner_timeout	<String>	Read-only	partner timeout value.

Operations

(click to see Properties)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

xen_lacp_stats

Configuration for shows runtime lacp stats resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
channel_id	<String>	Read-write	Channel ID if this interface represents a channel (LA/1, LA/2 ...).

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Maximum length = 50
member_stats	<xen_lacp_member_stats[]>	Read-write	Member Stats.
status	<String>	Read-only	LACP channel status.

Operations

(click to see Properties)

- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_lacp_stats **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_lacp_stats **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_lacp_stats resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_lacp_stats **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_lacp_stats resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_lacp_stats **?count=yes** Use this query-parameter to get the count of xen_lacp_stats resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_lacp_stats":[{

"channel_id":<String_value>,

"status":<String_value>,

"member_stats":[{

"partner_sync":<String_value>,

"actor_sys_priority":<Integer_value>,

"port_id":<String_value>,

"actor_collection":<String_value>,

"actor_port_priority":<Integer_value>,

"actor_rx_status":<String_value>,

"actor_select_status":<String_value>,

"actor_default":<String_value>,

"actor_sync":<String_value>,

"partner_distribution":<String_value>,

"actor_sys_mac":<String_value>,

"actor_port_no":<Integer_value>,

"partner_sys_mac":<String_value>,

"partner_port_priority":<Integer_value>,

"actor_distribution":<String_value>,

"partner_sys_priority":<Integer_value>,

"partner_key":<Integer_value>,
```

```
"actor_key":<Integer_value>,

"actor_mux_status":<String_value>,

"partner_collection":<String_value>,

"partner_aggregation":<String_value>,

"partner_default":<String_value>,

"partner_port_no":<Integer_value>,

"actor_timeout":<String_value>,

"partner_mode":<String_value>,

"actor_aggregation":<String_value>,

"actor_mode":<String_value>,

"port_status":<String_value>,

"partner_timeout":<String_value>}]}]}
```

xen_nsvpx_image

Configuration for Citrix ADC XVA File resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
			Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/xen_nsvpx_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_nsvpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_nsvpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_nsvpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_nsvpx_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_nsvpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_nsvpx_image **?count=yes** Use this query-parameter to get the count of xen_nsvpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_nsvpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,
```

```
"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_nsvpx_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_nsvpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,
```

```
"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/xen_nsvpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

xen_panwvpx_image

Configuration for XVA images of panw Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/xen_panwvpx_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_panwvpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_panwvpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_panwvpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_panwvpx_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_panwvpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_panwvpx_image **?count=yes** Use this query-parameter to get the count of xen_panwvpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_panwvpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_panwvpx_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_panwvpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/xen_panwvpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

xen_smx_gsbvpx_image

Configuration for XVA images of smx_gsb Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/xen_smx_gsbvpx_image/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_smx_gsbvpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_smx_gsbvpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_smx_gsbvpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_smx_gsbvpx_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_smx_gsbvpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_smx_gsbvpx_image **?count=yes** Use this query-parameter to get the count of xen_smx_gsbvpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_smx_gsbvpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_smx_gsbvpx_image/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_smx_gsbvpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/xen_smx_gsbvpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

xen_supplemental_pack

Configuration for XenServer Supplemental Pack resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
perform_reboot	<Boolean>	Read-write	Perform Appliance reboot automatically.
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
output	<String>	Read-only	Output of XenServer installation command.
act_id	<String>	Read-only	Activity Id.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- INSTALL
- DELETE

- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

install

URL: https://<MGMT-IP>/nitro/v2/config/xen_supplemental_pack/file_name_value<String>?action=install;onerror=<String_value> **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"xen_supplemental_pack": { }}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt}
```

delete

URL: https://<MGMT-IP>/nitro/v2/config/xen_supplemental_pack/file_name_value<String> **HTTPS Method:** DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_supplemental_pack **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_supplemental_pack **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_supplemental_pack resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_supplemental_pack **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_supplemental_pack resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_supplemental_pack **?count=yes** Use this query-parameter to get the count of xen_supplemental_pack resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_supplemental_pack":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"output":<String_value>,

"file_location_path":<String_value>,

"file_name":<String_value>,

"perform_reboot":<Boolean_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_supplemental_pack/file_name_value<String> **HTTPS Method:** GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_supplemental_pack":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"output":<String_value>,

"file_location_path":<String_value>,

"file_name":<String_value>,

"perform_reboot":<Boolean_value>,

"act_id":<String_value>,

"sync_operation":<Boolean_value>,

"file_last_modified_epoch":<Double_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/xen_supplemental_pack **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

xen_trend_microvpx_image

Configuration for XVA images of trend_micro Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE

- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.

- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/xen_trend_microvpx_image/file_name_value<String>
HTTPS Method: DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_trend_microvpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_trend_microvpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_trend_microvpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_trend_microvpx_image **?pagesize=#no;pageno=#no** Use this query-parameter to get the xen_trend_microvpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_trend_microvpx_image **?count=yes** Use this query-parameter to get the count of xen_trend_microvpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_trend_microvpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_trend_microvpx_image/file_name_value<String>
HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_trend_microvpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/xen_trend_microvpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

xen_upgrade

Configuration for Xen Upgrade resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
image_name	<String>	Read-write	image_name.
xen_version	<String>	Read-write	xen version.

Operations

(click to see Properties)

- UPGRADE
- GET (ALL)

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

To do this, you must specify the username and password in the request header of the NITRO request as follows:

X-NITRO-USER: <username>

X-NITRO-PASS: <password>

Note: In such cases, make sure that the request header DOES not include the following:

Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

upgrade

URL: https://<MGMT-IP>/nitro/v2/config/xen_upgrade/ **HTTPS Method:** PUT

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
{"xen_upgrade":{

"image_name":<String_value>,

"xen_version":<String_value>}}
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_upgrade":[{

"image_name":<String_value>,

"xen_version":<String_value>}]}
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_upgrade **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_upgrade **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_upgrade resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_upgrade **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_upgrade resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_upgrade **?count=yes** Use this query-parameter to get the count of xen_upgrade resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_upgrade":[{

"image_name":<String_value>,

"xen_version":<String_value>}]}
```

xen_websensevpx_image

Configuration for XVA images of websense Instance resource.

Properties

(click to see Operations)

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_location_path	<String>	Read-write	File Location on Client for upload/ download. Minimum length = 1
file_name	<String>	Read-write	File Name. Minimum length = 1 Maximum length = 256
file_size	<Integer>	Read-only	file_size.
file_last_modified	<String>	Read-only	Last Modified.

NAME	DATA TYPE	PERMISSIONS	DESCRIPTION
file_last_modified_epoch	<Double>	Read-only	Last Modified (Epoch).

Operations

(click to see Properties)

- UPLOAD
- DELETE
- GET (ALL)
- GET
- DOWNLOAD

Some options that you can use for each operations:

- **Getting warnings in response:** NITRO allows you to get warnings in an operation by specifying the 'warning' query parameter as 'yes'. For example, to get warnings while connecting to the NetScaler appliance, the URL is as follows:

  http:// <netscaler-ip-address> /nitro/v1/config/login?warning=yes

  If any, the warnings are displayed in the response payload with the HTTP code '209 X-NITRO-WARNING'.
- **Authenticated access for individual NITRO operations:** NITRO allows you to logon to the NetScaler appliance to perform individual operations. You can use this option instead of creating a NITRO session (using the login object) and then using that session to perform all operations,

  To do this, you must specify the username and password in the request header of the NITRO request as follows:

  X-NITRO-USER: <username>

  X-NITRO-PASS: <password>

  Note: In such cases, make sure that the request header DOES not include the following:

  Cookie:NITRO_AUTH_TOKEN= <tokenvalue>

`*Note:` * Mandatory parameters are marked in red and placeholder content is marked in green

The following parameters can be used in the nitro request : onerror <String_value>

Use this parameter to set the onerror status for nitro request. Applicable only for bulk requests.

Default value: EXIT

Possible values = EXIT, CONTINUE

delete

URL: https://<MGMT-IP>/nitro/v2/config/xen_websensevpx_image/file_name_value<String>
HTTPS Method: DELETE

Request Headers: X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt }
```

get (all)

URL: https://<MGMT-IP>/nitro/v2/config/xen_websensevpx_image **Query-parameters:**

filter https://<MGMT-IP>/nitro/v2/config/xen_websensevpx_image **?filter=property-name1:property-value1,property-name2:property-value2** Use this query-parameter to get the filtered set of xen_websensevpx_image resources configured on the system. You can set a filter on any property of the resource.

pagesize=#no;pageno=#no https://<MGMT-IP>/nitro/v2/config/xen_websensevpx_image **? pagesize=#no;pageno=#no** Use this query-parameter to get the xen_websensevpx_image resources in chunks.

count https://<MGMT-IP>/nitro/v2/config/xen_websensevpx_image **?count=yes** Use this query-parameter to get the count of xen_websensevpx_image resources.

HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_websensevpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

get

URL: https://<MGMT-IP>/nitro/v2/config/xen_websensevpx_image/file_name_value<String>
HTTPS Method: GET

Request Headers: Accept: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Headers:

Content-Type:application/json

Response Payload:

```
{ "errorcode": 0, "message": "Done", "severity": ;ltString_value;gt,
"xen_websensevpx_image":[{

"file_size":<Integer_value>,

"file_last_modified":<String_value>,

"file_last_modified_epoch":<Double_value>,

"file_location_path":<String_value>,

"is_dir":<Boolean_value>,

"file_name":<String_value>}]}
```

upload

URL: https://<MGMT-IP>/nitro/v2/upload/xen_websensevpx_image **HTTPS Method:** POST

Request Headers: Content-Type: application/json

X-NITRO-USER:username_value<String>

X-NITRO-PASS:password_value<String>

or

Cookie:NITRO_AUTH_TOKEN=token_value<String>

Request Payload:

```
Multi-part form data with File Stream
```

Response: HTTPS Status Code on Success: 200 OK HTTPS Status Code on Failure: 4xx (for general HTTPS errors) or 5xx (for NetScaler-MAS-specific errors). The response payload provides details of the error

Response Payload:

```
{ "errorcode": 0, "message": "Done }
```

Citrix SDX Nitro Error Messages

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_BR_UPGRADE	200023	Failed to upgrade Citrix SD-WAN Instance
ERR_UNKNOWN	10012	Request can not be processed
ERR_GUESTVM_INIT_CONFIG	400016	Failed to do initial configuration
ERR_ALREADY_EXISTS	10010	Already Exists
ERR_XAPI_RESET_NS_INTF	400020	Failed to reset Citrix ADC interfaces
ERR_JOIN_CLUSTER	30037	Failed to add VPX node to cluster
ERR_VPX_PART_OF_CLUSTER	30035	VPX Instance part of Cluster Instance
ERR_NS_RM_ULFDSERVER	60007	Failed to rm ULFD server
ERR_INVALID_PARAM	10008	Invalid Parameter
ERR_NS_GET_ROUTE	20021	Failed to get route
ERR_NS_GET_VSERVER_APPFLOW_CONFIG	20042	Failed to get vserver appflow configuration from Citrix ADC
ERR_SFTP_DOWNLOAD	16003	SFTP failed to download
ERR_INSTANCE_UNMANAGED	50029	Manage Instance Error

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_XAPI_CREATE_NETWORK	150016	Failed to create Network
ERR_XAPI_RESET_VM	400021	Failed to reset VM
ERR_SCP_DOWNLOAD	16001	SCP failed to upload
ERR_XAPI_GET_SR	150018	Failed to get SR
ERR_SDX_UPGRADE	50005	Failed to upgrade SDX
ERR_DELETE_BOND	40012	Error while deleting bond on XenServer
ERR_NS_GET_HOSTNAME	20017	Failed to get hostname
ERR_CONFIG_INPROGRESS	11001	Configuration in progress
ERR_BR_BROKER_GET_LINK	30040	Failed to Get Link Configuration from Citrix SD-WAN Accelerator
ERR_XAPI_ADD_VBD	150020	Failed to add VBD
ERR_NS_RM_ROUTE	20009	Failed to remove route
ERR_CREATE_BOND	40011	Error while creating bond on XenServer
ERR_NS_SET_HOSTNAME	20006	Failed to set hostname
ERR_NS_ADD_ULFDSERVER	60006	Failed to add ULFD server
ERR_SDWANVW_CONFIGURE_APPFLOW	50027	Failed to Configure Appflow on Citrix SD-WAN VW
ERR_BR_REMOTELICENSE	20036	Failed to Configure Remote License Server on Citrix SD-WAN

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_MISSING_ACTION	10004	Action not specified for this request
ERR_NS_SAVE_CONFIG	20015	Failed to save configuration
ERR_GUESTVM_BACKUP_CONFIG	40005	Failed to backup configuration
ERR_NS_GET_HANODE	20019	Failed to get ha node
ERR_REBOOT	11010	Failed to reboot
ERR_CBWANOPT_UPGRADE	50026	Failed to upgrade Citrix SD-WAN WO
ERR_BR_CONFIGURE_APPFLOW	50006	Failed to Configure Appflow on Citrix SD-WAN
ERR_XAPI_GET_TASK	15006	Failed to get Task
ERR_GET_SDX_CONFIG	500001	Failed to get SDX configuration
ERR_BIND_UNBIND_CHANNEL_PROPERTIES	40010	Error while doing bind-unbind for a channel in VM
ERR_BACKUP	30030	Error in backup operation
ERR_XAPI_GET_PIF	15005	Failed to get PIF
ERR_NS_INSTALL_CERT	20002	Failed to install SSL certificates
ERR_INVALID_POLICY	15014	Invalid policy
ERR_INVALID_PORT	15010	Invalid port
ERR_PORT_SETTING_FAILED	15013	Failed to modify port settings
ERR_NS_SET_L2VLANCONFIG	20044	Failed to set L2VLAN config

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_SCP_UPLOAD	16000	SCP failed to upload
ERR_SDWANVW_GET_APPFLOW_CONFIGURATION	50028	Failed to Get Appflow configuration information on Citrix SD-WAN VW
ERR_SSH_REQUEST	16004	SSH failed to execute request
ERR_DOCKERAPI_CONTAINER_START	50024	Failed to start docker container
ERR_NS_SNMP_CONFIG	20045	Failed to do SNMP configuration
ERR_GUESTVM_UPGRADE_TOOLS	40002	Failed to upgrade sdxtools
ERR_XAPI_ADVVMS	150026	Failed to Advertise VM UUID
ERR_SVM_REBOOT	20000	Failed to reboot Management Service
ERR_XAPI_DEL_VBD	150021	Failed to delete VBD
ERR_SEND_INVENTORY_MESSAGE	500004	Failed to send message to inventory
ERR_XAPI_GET_VBD	150019	Failed to get VBD
ERR_BR_CONFIGSYSLOG	20030	Failed to Configure Syslog server on Citrix SD-WAN
ERR_PARAM_MISMATCH	10016	Build and Documentation file version mismatch
ERR_BR_ENABLE	20027	Failed to enable Citrix SD-WAN
ERR_FORM_CLUSTER	30036	Failed to form cluster on VPX
ERR_NS_SNMP_TRAP_CONFIG	20046	Failed to do SNMP trap configuration

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_NOT_AUTHORIZED	10013	Not authorized to perform this operation
ERR_ADD_TRAP	500003	Failed to add trap destination
ERR_ADD_DEVICE	30032	Failed to Add Device
ERR_DOCKERAPI_CONTAINER_CREATE	50023	Failed to create docker container
ERR_PARAM_MISSING	10007	Parameter missing
ERR_RESOURCE_CONVERSION	10002	Resource conversion failed
ERR_BR_BROKER_UPGRADE	20100	Failed to upgrade Citrix SD-WAN Accelerator
ERR_MAX_BRVPX	11011	Maximum Citrix SD-WAN Instances reached
ERR_GUESTVM_DOWNGRADE	400015	Failed to downgrade
ERR_XAPI_ADD_VM	15001	Failed to add VM
ERR_MAX_NSVPX	11002	Maximum Citrix ADC Instances reached
ERR_SERVICE_UNAVAILABLE	60010	Service is not available
ERR_ADD_CHANNEL_PROPERTIES	40009	Error while adding channel to VM
ERR_XAPI_DELETE_NETWORK	150017	Failed to delete Network
ERR_GUESTVM_TECH_SUPPORT	40006	Error getting techsupport bundle

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_NS_GET_RUNNINGCONFIG	20003	Failed to get running configuration
ERR_XAPI_GET_VM	15000	Failed to get VM
ERR_ENABLING_CLUSTER	30039	Failed to enable cluster feature on VPX
ERR_MANDATORY_PARAM_MISSING	10006	Mandatory parameter missing
ERR_BULK_OPERATION	30000	Error in bulk operation
ERR_DISK_STOP_LOCATE_FAILED	20202	Failed to stop disk locate
ERR_SET_STATUS_CALLHOME	20052	Failed to enable/disable CallHome in Citrix ADC Instance
ERR_NS_SET_NSCONFIG	20008	Failed to set Citrix ADC config
ERR_BR_GET_BRCONFIG	20024	Failed to get Citrix SD-WAN Instance config
ERR_DISK_FORMAT_FAILED	20200	Failed to format disk
ERR_NS_GET_STATNS	20022	Failed to get statistics of Citrix ADC Instance
ERR_NOT_EXISTS	10011	Resource does not exist
ERR_NTP_SERVER_ADD	50011	Failed to add NTP Server
ERR_CLU_NOT_ENABLED	30034	Cluster feature not enabled on VPX
ERR_NS_RM_INTERNAL_ARP	20054	Failed to remove internal arp

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_BACKUPRESTORE_IN PROGRESS	11012	System maintenance in progress. Please try after some time
ERR_RM_FORWARDINGSE SSION	20038	Failed to delete Forwarding Sessions on Citrix ADC Instance
ERR_GUESTVM_SET_LICE NSE	40003	Failed to set license
ERR_XAPI_TECHSUPPORT	15009	Failed to generate Technical support archive
ERR_GUESTVM_RESTORE _CONFIG	40004	Failed to retore configuration
ERR_APPLY_LICENSES	11004	Failed to apply licenses
ERR_DOCKERAPI_CONTAI NER_DELETE	50020	Failed to delete docker container
ERR_NO_THROUGHOUT	11006	Throughput can not be assigned, Please get the licenses.
ERR_SFTP_UPLOAD	16002	SFTP failed to upload
ERR_SESSION_TIMEOUT_ VALIDATION_FAILED	10019	Invalid login session timeout
ERR_NOT_SUPPORTED_IN _THIS_VERSION_OF_XEN	10017	Feature not supported for this version of XenServer
ERR_XAPI_GET_VDI	150022	Failed to get VDI
ERR_BR_CONFIGALERT	20029	Failed to Configure Alert on Citrix SD-WAN
ERR_DOCKERAPI_CONTAI NERID_NOTFOUND	50021	Failed to find container ID on host

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_BR_CONFIGBANDWIDTH	20031	Failed to Configure Bandwidth on Citrix SD-WAN
ERR_NS_GET_SAVEDCONFIG	20004	Failed to get saved configuration
ERR_NS_GET_NSCONFIG	20016	Failed to get Citrix ADC config
ERR_XAPI_ADD_VDI	150023	Failed to add VDI
ERR_ADD_DOMAINID	500032	Failed to add observation domain ID
ERR_XAPI_SET_LOOPBACK	400022	Failed to configure loopback settings
ERR_SVM_UPGRADE	20001	Failed to upgrade Management Service
ERR_NS_ADD_SET_NSIP	20011	Failed to add/set nsip
ERR_NS_ADD_SET_SYS_CMDPOLICY	20013	Failed to add/set system command policy
ERR_XAPI_ADD_VLAN	400018	Failed to add VLAN
ERR_SESSION_NOT_EXIST	444	Session expired or cancelled. Please login again
ERR_INVALID_RESOURCE	10005	Invalid Resource
DONE	0	Done
ERR_XAPI_GET_CPU	15002	Failed to get CPU
ERR_LOGIN_FAILED	10001	Invalid username/password
ERR_NS_BIND_SYS_USER	20014	Failed to bind system user
ERR_GET_BOND	40013	Error while getting bonds from XenServer

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_REMOVE_CLUSTER	30038	Failed to remove cluster instance from VPX
ERR_NS_ADD_SET_ROUTE	20010	Failed to add/set route
ERR_PLUGIN_CALL_FAILED	10018	Plugin call failed
ERR_SYSTEM_NOT_INITIALIZED	50016	System not initialized
ERR_BR_ADDUSER	20032	Failed to Add User on Citrix SD-WAN
ERR_DOCKERAPI_CONTAINER_REBOOT	50025	Failed to reboot docker container
ERR_XAPI_DEL_VIF	150015	Failed to delete VIF
ERR_NS_GET_NSIP	20018	Failed to get Citrix ADC Instance IP Address
ERR_BR_LOCALLICENSE	20035	Failed to Configure Local License Server on Citrix SD-WAN
ERR_BR_RESTART	20033	Failed to Restart Citrix SD-WAN
ERR_SET_CHANNEL_PROPERTIES	40007	Error while setting channel properties
ERR_UNLOCK_INVENTORY	50010	Failed to unlock inventory
ERR_GUESTVM_UPGRADE	40001	Failed to upgrade
ERR_NS_UPGRADE	20005	Failed to upgrade Citrix ADC Instance
ERR_XAPI_ADD_NSTRACE	150025	Failed to add Citrix ADC trace uuid

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_NTP_SERVER_MODIFY	50012	Failed to modify NTP Server
ERR_GET_CBWANOPT_CONFIG	50031	Failed to get Citrix SD-WAN WO configuration
ERR_INVALID_ACTION	10015	Invalid Action
ERR_BR_SNMP_ENABLE	20048	Failed to enable SNMP in Citrix SD-WAN
ERR_EXPORT_REPORT	50008	Failed to Export report
ERR_XAPI_GET_VLAN	400017	Failed to get VLAN
ERR_INVALID_HOST_TYPE	10003	Invalid host type
ERR_GET_SDWANVW_CONFIG	50030	Failed to get Citrix SD-WAN VW configuration
ERR_MIGRATION_IN_PROGRESS	50017	Migration In Progress
ERR_ADD_MANAGER_BR	20026	Failed to add SNMP Manager to Citrix SD-WAN Instance
ERR_XAPI_VIRTUAL_NMI	150027	Failed to generate Crash dump on Citrix ADC VPX
ERR_CBSDX_UPGRADE	50022	Failed to upgrade CBSDX
ERR_ADD_TRAP_BR	20050	Failed to add SNMP trap destination in Citrix SD-WAN.
ERR_MODIFICATION_INPROGRESS	11009	Modification in progress
ERR_DEVICE_NOT_FOUND	10009	Device not found
ERR_NS_SET_VSERVER_APPFLOW_CONFIG	20043	Failed to set vserver appflow configuration at Citrix ADC

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_NS_GET_NSSSLCERT	20040	Failed to get SSL certificates from Citrix ADC Instance
ERR_XAPI_GET_NETWORK	15004	Failed to get Network
ERR_XAPI_GET_HOST	15003	Failed to get Host
ERR_MAX_THROUGHOUT	11008	Maximum Throughput reached
ERR_SET_CALLHOME	20051	Failed to configure CallHome in Citrix ADC Instance
ERR_NS_ADD_SET_SNMP_COMMUNITY	20007	Failed to add/set snmp community
ERR_LOCK_INVENTORY	50009	Failed to lock inventory
ERR_GET_NS_CHANNEL_INFO	40014	Error while getting channel information from Citrix ADC
ERR_GUESTVM_RE_INIT	400019	Failed to reinitialize configuration
ERR_NS_GET_CONFIG_DIFF	20047	Failed to get configuration diff from Citrix ADC Instance
ERR_BR_DISABLE	20028	Failed to disable Citrix SD-WAN
ERR_CLU_NOT_SUPPORTED	30033	Cluster not supported on VPX
ERR_BR_RESTORECONFIG	20034	Failed to Restore Config on Citrix SD-WAN
ERR_SSL_HANDSHAKE_FAILURE	50015	SSL Handshake Failed
ERR_REMOVE_CHANNEL_PROPERTIES	40008	Error while removing channel from VM

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_NS_RM_SSL_CERTKEY	20041	Failed to remove ssl certkey from Citrix ADC Instance
ERR_BACKUP_PARTIAL	30031	Backup partially completed
ERR_BR_SFTP_ENABLE	20049	Failed to enable SFTP in Citrix SD-WAN
ERR_NTP_SERVER_DELETE	50013	Failed to delete NTP Server
ERR_NS_ADD_SET_INTERNAL_ARP	20055	Failed to add/set internal arp
ERR_BR_GET_APPFLOW_CONFIGURATION	50007	Failed to Get Appflow configuration information on Citrix SD-WAN
ERR_NS_ADD_SET_SYS_USER	20012	Failed to add/set system user
ERR_BR_LOGINTOKEN	20039	Failed to get Session ID on Citrix SD-WAN Instance
ERR_XAPI_GET_VIF	15007	Failed to get VIF
ERR_MAX_SSLCORES	11003	Maximum SSL cores reached
ERR_ADD_FORWARDINGSESSION	20037	Failed to add Forwarding Sessions on Citrix ADC Instance
ERR_DEFAULT_ENTITY	11005	Default entity can not be modified/deleted
ERR_NS_SET_MODE	20053	Failed to set mode
ERR_DISK_LOCATE_FAILED	20201	Failed to locate disk
ERR_NS_CERT_CREATE_FAILED	20203	Failed to create certificate

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_BR_REPLICATE_CONFIG	20025	Failed to replicate Citrix SD-WAN Instance config
ERR_NOT_FOUND_IN_SELECTED_DURATION	60009	No result found in selected duration.
ERR_NITRO_BLOCKED	50018	Nitro access blocked
ERR_PORT_DETAILS_FAILED	15011	Failed to get port details
ERR_APPLYING_CONFIG_FAILED	60009	Failed to applying config.
ERR_NS_GET_NSVERSION	20020	Failed to get Citrix ADC instance version
ERR_NO_VSERVER_LICENSE	60008	Selected Vserver Marked as UnLicensed
ERR_TIMEZONE_SET	50014	Failed to set timezone
ERR_DEVICE_NOT_REACHABLE	500002	Failed to reach device
ERR_PORT_RESTORE_DEFAULT_FAILED	15012	Failed to restore defaults for port
ERR_DECIMAL_THROUGHOUT	11007	Throughput can not be assigned in decimals
ERR_VALIDATION_FAILED	10014	Validation failed
ERR_CHANGE_PASSWORD_SVM	11000	Fail to change password
ERR_ADM_UPGRADE	20001	Failed to upgrade ADM
ERR_XAPI_DEL_VDI	150024	Failed to delete VDI
ERR_XAPI_ADD_VIF	15008	Failed to add VIF

ERROR CODE	ERROR CODE	ERROR MESSAGE
ERR_DOCKERAPI_CONTAINER_STOP	50019	Failed to stop docker container