



# **Cisco Prime Collaboration Assurance 11.5 North Bound Interface Guide**

## Contents



Preface .....	4
Conventions .....	4
Chapter 1: Overview of NBI Service .....	5
Chapter 2: Resources .....	9
<a href="https://&lt;ipaddress&gt;/emsam/nbi/callQuality">https://&lt;ipaddress&gt;/emsam/nbi/callQuality</a> .....	9
<a href="https://&lt;ipaddress&gt;/emsam/nbi/customer">https://&lt;ipaddress&gt;/emsam/nbi/customer</a> .....	9
<a href="https://&lt;ipaddress&gt;/emsam/nbi/session">https://&lt;ipaddress&gt;/emsam/nbi/session</a> .....	11
<a href="https://&lt;ipaddress&gt;/emsam/nbi/device">https://&lt;ipaddress&gt;/emsam/nbi/device</a> .....	12
<a href="https://&lt;ipaddress&gt;/emsam/nbi/device/clusterEndpointCount">https://&lt;ipaddress&gt;/emsam/nbi/device/clusterEndpointCount</a> .....	15
<a href="https://&lt;ipaddress&gt;/emsam/nbi/device/customer">https://&lt;ipaddress&gt;/emsam/nbi/device/customer</a> .....	15
<a href="https://&lt;ipaddress&gt;/emsam/nbi/device/deleteDevices">https://&lt;ipaddress&gt;/emsam/nbi/device/deleteDevices</a> .....	16
<a href="https://&lt;ipaddress&gt;/emsam/nbi/device/suspend">https://&lt;ipaddress&gt;/emsam/nbi/device/suspend</a> .....	17
<a href="https://&lt;ipaddress&gt;/emsam/nbi/device/resume">https://&lt;ipaddress&gt;/emsam/nbi/device/resume</a> .....	17
<a href="https://&lt;ipaddress&gt;/emsam/nbi/nbidoc">https://&lt;ipaddress&gt;/emsam/nbi/nbidoc</a> .....	18
<a href="https://&lt;ipaddress&gt;/emsam/nbi/phone">https://&lt;ipaddress&gt;/emsam/nbi/phone</a> .....	18
<a href="https://&lt;ipaddress&gt;/emsam/nbi/callFailure">https://&lt;ipaddress&gt;/emsam/nbi/callFailure</a> .....	19
<a href="https://&lt;ipaddress&gt;/emsam/nbi/endpoint">https://&lt;ipaddress&gt;/emsam/nbi/endpoint</a> .....	19
<a href="https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/start">https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/start</a> .....	20
<a href="https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/report/{tsSessionId}">https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/report/{tsSessionId}</a> .....	21
<a href="https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/stop">https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/stop</a> .....	21
<a href="https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/status/{tsSessionId}">https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/status/{tsSessionId}</a> .....	22
<a href="https://&lt;ipaddress&gt;/emsam/nbi/event">https://&lt;ipaddress&gt;/emsam/nbi/event</a> .....	22
<a href="https://&lt;ipaddress&gt;/emsam/nbi/nbiLicense">https://&lt;ipaddress&gt;/emsam/nbi/nbiLicense</a> .....	23
<a href="https://&lt;ipaddress&gt;/emsam/nbi/alarm">https://&lt;ipaddress&gt;/emsam/nbi/alarm</a> .....	24
<a href="https://&lt;ipaddress&gt;/emsam/nbi/alarm/operate">https://&lt;ipaddress&gt;/emsam/nbi/alarm/operate</a> .....	24
<a href="https://&lt;ipaddress&gt;/emsam/nbi/user">https://&lt;ipaddress&gt;/emsam/nbi/user</a> .....	25
<a href="https://&lt;ipaddress&gt;/emsam/nbi/user/password">https://&lt;ipaddress&gt;/emsam/nbi/user/password</a> .....	27
<a href="https://&lt;ipaddress&gt;/emsam/nbi/user/customer">https://&lt;ipaddress&gt;/emsam/nbi/user/customer</a> .....	28
<a href="https://&lt;ipaddress&gt;/emsam/nbi/credential">https://&lt;ipaddress&gt;/emsam/nbi/credential</a> .....	29

<https://<ipaddress>/emsam/nbi/diagnosticstest> ..... 31

<https://<ipaddress>/emsam/nbi/nbiDocumentation> ..... 32

# Preface

This manual describes the Cisco Prime Collaboration Provisioning northbound interfaces. It provides instructions for using and administering it.

## Conventions

This document uses the conventions as shown in the following table:

<b>Item</b>	<b>Convention</b>
<b>Commands and keywords</b>	<b>boldface font</b>
<b>Variables for which you supply values</b>	<i>italic font</i>
<b>Information you enter</b>	<b>boldface screen font</b>

# Chapter 1: Overview of NBI Service

Resource	Method	Description
<a href="https://&lt;ipaddress&gt;/emsam/nbi/callQuality">https://&lt;ipaddress&gt;/emsam/nbi/callQuality</a>	<a href="#">GET</a>	API to get number of poor calls and total calls per cluster for each location for past 1 hour.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/customer">https://&lt;ipaddress&gt;/emsam/nbi/customer</a>	<a href="#">PUT</a>	API to add/edit customer.
	<a href="#">GET</a>	Returns the customer if exists.
	<a href="#">POST</a>	This API will return all the customers that match the filtering criteria.
	<a href="#">DELETE</a>	API to delete a customer.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/session">https://&lt;ipaddress&gt;/emsam/nbi/session</a>	<a href="#">POST</a>	API to get all sessions, user can use different filter criteria and get only required sessions.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/device">https://&lt;ipaddress&gt;/emsam/nbi/device</a>	<a href="#">PUT</a>	API to add/edit devices.
	<a href="#">DELETE</a>	API to delete a device.
	<a href="#">POST</a>	This API will return all the device that match the filtering criteria.
	<a href="#">GET</a>	Returns the device if exists.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/device/clusterEndpointCount">https://&lt;ipaddress&gt;/emsam/nbi/device/clusterEndpointCount</a>	<a href="#">POST</a>	API to get CUCM Cluster level Endpoint count. Given IP Address of CUCM (publisher or subscriber), It will return CUCM's cluster level count of registered, unregistered endpoints and cluster name.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/device/customer">https://&lt;ipaddress&gt;/emsam/nbi/device/customer</a>	<a href="#">POST</a>	API to update the customer association of device(s). API requires customer name and comma separated list of IP addresses as input. If the IP address is of a CUCM or VCS then the customer association for all the registered devices will also be updated. If the customer name is provided as blank, then the device is removed from that customer domain.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/device/deleteDevices">https://&lt;ipaddress&gt;/emsam/nbi/device/deleteDevices</a>	<a href="#">POST</a>	API to delete multiple device.

<a href="https://&lt;ipaddress&gt;/emsam/nbi/device/suspend">https://&lt;ipaddress&gt;/emsam/nbi/device/suspend</a>	<a href="#">POST</a>	API to suspend a device.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/device/resume">https://&lt;ipaddress&gt;/emsam/nbi/device/resume</a>	<a href="#">POST</a>	API to resume management of a device.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/nbidoc">https://&lt;ipaddress&gt;/emsam/nbi/nbidoc</a>	<a href="#">GET</a>	
<a href="https://&lt;ipaddress&gt;/emsam/nbi/phone">https://&lt;ipaddress&gt;/emsam/nbi/phone</a>	<a href="#">GET</a>	
<a href="https://&lt;ipaddress&gt;/emsam/nbi/callFailure">https://&lt;ipaddress&gt;/emsam/nbi/callFailure</a>	<a href="#">GET</a>	API to get number of failed calls and total calls per cluster for each location for past 1 hour.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/endpoint">https://&lt;ipaddress&gt;/emsam/nbi/endpoint</a>	<a href="#">POST</a> <a href="#">GET</a>	
<a href="https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/start">https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/start</a>	<a href="#">PUT</a>	API to start a troubleshooting session.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/report/{tsSessionId}">https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/report/{tsSessionId}</a>	<a href="#">GET</a>	API to get the report for a troubleshooting session.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/stop">https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/stop</a>	<a href="#">PUT</a>	API to stop a troubleshooting session.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/status/{tsSessionId}">https://&lt;ipaddress&gt;/emsam/nbi/troubleshoot/status/{tsSessionId}</a>	<a href="#">GET</a>	API to get the status of a troubleshooting session.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/event">https://&lt;ipaddress&gt;/emsam/nbi/event</a>	<a href="#">POST</a>	This API will return all the alarms that match the filtering criteria.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/nbiLicense">https://&lt;ipaddress&gt;/emsam/nbi/nbiLicense</a>	<a href="#">GET</a>	API to retrieve the license type, license counts and consumed license counts.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/alarm">https://&lt;ipaddress&gt;/emsam/nbi/alarm</a>	<a href="#">POST</a>	This API will return all the device that match the filtering criteria.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/alarm/operate">https://&lt;ipaddress&gt;/emsam/nbi/alarm/operate</a>	<a href="#">POST</a>	API to perform CLEAR/ACKNOWLEDGE/UNACKNOWLEDGE/AN NOTATE operations on alarms.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/user">https://&lt;ipaddress&gt;/emsam/nbi/user</a>	<a href="#">GET</a>	Returns the user if exists.

	<p><a href="#">PUT</a></p> <p><a href="#">DELETE</a></p> <p><a href="#">POST</a></p>	<p>API for adding new user or updating existing user. API support one or more user addition/update. Different user role values are ['REPORTVIEWER','HELPDESK', 'OPERATOR', 'NETWORKADMIN','SYSTEMADMIN','SUPERADMIN']</p> <p>NOTE : While adding any higher ROLE to PC through API, ensure adding all the LOWER ROLES explicitly.</p> <p>For example, make sure while adding user with role NETWORKADMIN explicitly provide below roles like REPORTVIEWER, HELPDESK and OPERATOR. API expect caller should take care of passing all applicable roles when adding user.</p> <p>API to delete a user.</p> <p>This API will return all the users that matches the input user names.</p>
<p><a href="https://&lt;ipaddress&gt;/emsam/nbi/user/password">https://&lt;ipaddress&gt;/emsam/nbi/user/password</a></p>	<p><a href="#">POST</a></p>	<p>API to change password for a user.</p> <p>Password should match below mentioned password rules:</p> <ol style="list-style-type: none"> <li>1. Password should be minimum 8 characters and Maximum 20 character.</li> <li>2. Password must contain at least one of each - lowercase letter, uppercase letter, number, and special character.</li> <li>3. No character in the password can be repeated more than three times consecutively.</li> <li>4. Password cannot contain non-ASCII characters.</li> <li>5. Password cannot be the same as the username, and cannot be the username reversed.</li> <li>6. Password cannot or contain "cisco", "ocsic"</li> </ol>
<p><a href="https://&lt;ipaddress&gt;/emsam/nbi/user/customer">https://&lt;ipaddress&gt;/emsam/nbi/user/customer</a></p>	<p><a href="#">POST</a></p>	<p>This API to replace customer assigned for a given user. API support one or more user customer update.</p>
<p><a href="https://&lt;ipaddress&gt;/emsam/nbi/credential">https://&lt;ipaddress&gt;/emsam/nbi/credential</a></p>	<p><a href="#">POST</a></p> <p><a href="#">PUT</a></p> <p><a href="#">GET</a></p>	<p>This API will return all the credential objects that match the filtering criteria.</p> <p>API to add/edit credentials.</p> <p>Returns the credential if exists.</p>

	<a href="#"><u>DELETE</u></a>	API to delete a credential.
<a href="https://&lt;ipaddress&gt;/emsam/nbi/diagnostictest"><u>https://&lt;ipaddress&gt;/emsam/nbi/diagnostictest</u></a>	<a href="#"><u>PUT</u></a> <a href="#"><u>POST</u></a>	
<a href="https://&lt;ipaddress&gt;/emsam/nbi/nbiDocumentation"><u>https://&lt;ipaddress&gt;/emsam/nbi/nbiDocumentation</u></a>	<a href="#"><u>GET</u></a>	

# Chapter 2: Resources

This chapter describes the Prime Collaboration Assurance NBI services.

## <https://<ipaddress>/emsam/nbi/callQuality>

This NBI service fetches Call Quality (Poor calls) details

### Methods

<b>GET</b>	getPoorCalls ()
------------	-----------------

API to get number of poor calls and total calls per cluster for each location for past 1 hour.

**request**  
unspecified

**responses**

**status:**  
200 - OK

**representations**

<b>application/json</b>	Example: The output of this API is a JSON that will contain the overall result of the operation.  The response will contain result per cluster per location.  <a href="#">Sample response</a>
-------------------------	--

## <https://<ipaddress>/emsam/nbi/customer>

This NBI service handles various operations on customer operations like adding, editing, deleting customer.

Also retrieval of customer based on name and other various filtering criteria.

[Customer specific error codes and Messages](#)

### Methods

<b>PUT</b>	omer ()
------------	---------

API to add/edit customer.

**request**  
**representations**

<b>application/json</b>	Example: This API takes JSON as input. JSON can contain multiple customers (combination of customers to add and modify).
-------------------------	---

[Sample Input](#)

**responses**

**status:**

200 - OK

**representations**

**application/json**

Example:

The output of this API is a JSON that will contain the overall result of the operation, values can be  
SUCCESS - if the operation on all the devices was successful.  
FAILURE - if the operation on all the devices failed.  
PARTIALLY SUCCESSFUL - if the operation succeeded on some of the devices.

The response will also contain result per customer which will contain the status as  
SUCCESS - if the customer was added successfully or not.  
FAILURE - In the event of a failure to add a customer the error code and message fields will also be populated.

[Sample response](#)

**GET**

getCustomer ()

Returns the customer if exists.

**request**

**query params**

name	<a href="#">string</a>
id	<a href="#">long</a> default: 0

**responses**

**status:**

200 - OK

**representations**

**application/json**

Example:

[Sample response](#)

**POST**

getCustomers ()

This API will return all the customers that match the filtering criteria.

**request**

**representations**

<b>application/json</b>	<p>Example:  The <b>filter</b> and <b>sort</b> fields are not mandatory.  The output of the API will be limited to 500 records.  This API will need to be called multiple times providing data for the <b>pagination</b> field to retrieve more data.  <a href="#">Sample input data</a></p>
-------------------------	--

**responses**

**status:**

200 - OK

[representations](#)

<b>application/json</b>	<p>Example:  The response json will be similar to that of getDevice operation but with a list of devices.  <a href="#">Sample response data</a></p>
-------------------------	---

<b>DELETE</b>	deleteCustomer ()
---------------	-------------------

API to delete a customer.

**request**

[query params](#)

<b>name</b>	<a href="#">string</a>
<b>id</b>	<a href="#">long</a> default: 0
<b>deleteDevices</b>	<a href="#">boolean</a> default: false

**responses**

**status:**

200 - OK

[representations](#)

<b>application/json</b>	<p>Example:  The response will contain a json object similar to the response of add customer API.  <a href="#">Sample response</a></p>
-------------------------	--

## https://<ipaddress>/emsam/nbi/session

Session Monitoring nbi services.

**Methods**

**POST** AllSessions ()

API to get all sessions, user can use different filter criteria and get only required sessions.

**request**  
[representations](#)

<b>*/*</b>	Example: This API can be used get all sessions based on filter criteria provided. <a href="#">Sample Input</a>
------------	--

**responses**  
**status:**  
200 - OK  
[representations](#)

<b>application/json</b>	Example: <a href="#">Sample response</a>
-------------------------	---

## https://<ipaddress>/emsam/nbi/device

This NBI service handles various operations on devices, operations like adding, editing, deleting devices, suspending and resuming management of devices.

Also retrieval of devices based on ipaddress and other various complex filtering criteria.

[Generic error codes and Messages](#)

[Device specific error codes and Messages](#)

### Methods

**PUT** addDevice ()

API to add/edit devices.

**request**  
[representations](#)

<b>application/json</b>	Example: This API takes JSON as input. JSON can contain multiple devices (combination of devices to add and modify). If the device should be discovered using a credential profile set the useProfile field to true in the input json, for this credential profiles should have been pre-created in CPCM. Or one can provide the entire credential set per device using the field credential in the input json which represents a credential map, refer sample input. One can also provide the id of a pre-created per device credential in the json with credentialId field. If the PC deployment is in MSP mode and the device to be managed is reachable in the network with a private IP address
-------------------------	---

then the same can be provided using field `privateIp`.  
 When a private IP is provided in the input then customer name is a mandatory field to associate the device to a customer.  
 The private dns field can be used to provide the private hostname of the device.  
 Customer name can be provided using the input field `admGroupName`.  
 To disable logical discovery when adding a seed device like CUCM, TMS, VCS set the **enableLogicalDiscovery** flag to **false**.  
 Note: Private IP, Customer name will be considered only if the PC deployment is in MSP mode.  
 In case of edit devices, API allows only editing the credentials as rest of the device attributes are dynamically populated during discovery process.  
 In order to change customer association of a device use **/device/customer** url, since there is no need to re-discover the devices.  
 To know what credentials are mandatory for a particular device type refer [Doc Wiki](#) .  
 For certain devices like VCenter which does not have an associated credentials, provide some junk value as SNMP Read CS because SNMP Read CS is a mandatory credential input for any device.  
[Sample Input](#)

## responses

### status:

200 - OK

### representations

<b>application/json</b>	<p>Example:          The output of this API is a JSON that will contain the overall result of the operation, values can be          SUCCESS - if the operation on all the devices was successful.          FAILURE - if the operation on all the devices failed.          PARTIALLY SUCCESSFUL - if the operation succeeded on some of the devices.</p> <p>The response will also contain result per device which will contain the status as          SUCCESS - if the device was added successfully and a discovery job is created.          FAILURE - In the event of a failure to add a device the error code and message fields will also be populated.  <a href="#">Sample response</a></p>
<b>DELETE</b>	deleteDevice ()

API to delete a device.

## request

### query params

id	<a href="#">long</a>
----	----------------------

### responses

#### status:

200 - OK

#### representations

application/json	Example: The response will contain a json object similar to the response of add device API. <a href="#">Sample response</a>
POST	getDevices ()

This API will return all the device that match the filtering criteria.

### request

#### representations

application/json	Example: The <b>filter</b> and <b>sort</b> fields are not mandatory. The output of the API will be limited to 500 records. This API will need to be called multiple times providing data for the <b>pagination</b> field to retrieve more data. <a href="#">Sample input data</a>
------------------	---

### responses

#### status:

200 - OK

#### representations

application/json	Example: The response json will be similar to that of getDevice operation but with a list of devices. <a href="#">Sample response data</a>
GET	getDevice ()

Returns the device if exists.

### request

#### query params

ipAddress	<a href="#">string</a>
id	<a href="#">long</a>

### responses

#### status:

200 - OK  
[representations](#)

<b>application/json</b>	Example: <a href="#">Sample response</a>
-------------------------	---

## <https://<ipaddress>/emsam/nbi/device/clusterEndpointCount>

### Methods

<b>POST</b>	getClusterEndpointCount ()
-------------	----------------------------

API to get CUCM Cluster level Endpoint count.  
Given IP Address of CUCM (publisher or subscriber), it will return CUCM's cluster level count of registered, unregistered endpoints and cluster name.

### request [representations](#)

<b>application/json</b>	Example: This API takes JSON as input. JSON should contain "ipAddress" field specifying IP Address(s) of CUCM. IP Address can be specified in comma separated format if there are multiple CUCM. <a href="#">Sample Input</a>
-------------------------	--

### responses **status:** 200 - OK [representations](#)

<b>application/json</b>	Example: The output of this API is a JSON that will contain the overall result of the operation, values can be SUCCESS - if the request on all the CUCM(s) passed was successful. FAILURE - if the operation on all the CUCM(s) failed. PARTIALLY SUCCESSFUL - if the operation succeeded on some of the CUCM(s).  The response will also contain result per CUCM which will contain the status as SUCCESS - If successfully retrieved the required information. FAILURE - In the event of a failure error code and message fields will also be populated. <a href="#">Sample response</a>
-------------------------	---

## <https://<ipaddress>/emsam/nbi/device/customer>

## Methods

**POST** updateCustomerAssociation ()

API to update the customer association of device(s).

API requires customer name and comma separated list of IP addresses as input.

If the IP address is of a CUCM or VCS then the customer association for all the registered devices will also be updated.

If the customer name is provided as blank, then the device is removed from that customer domain.

### request

#### representations

##### application/json

Example:

This API takes JSON as input. JSON should contain "ipAddress" field specifying IP Address(s) of device(s).

IP Address can be specified in comma separated format if there are multiple devices.

[Sample Input](#)

### responses

#### status:

200 - OK

#### representations

##### application/json

Example:

The output of this API is a JSON that will contain the overall result of the operation, values can be

SUCCESS - if the request on all the devices was successful.

FAILURE - if the operation on all the devices failed.

[Sample response](#)

## https://<ipaddress>/emsam/nbi/device/deleteDevices

## Methods

**POST** deleteDevices ()

API to delete multiple device.

### request

#### representations

##### application/json

Example:

This API can be used to delete single or multiple devices.

A json with an array of device ipaddress should be provided as input.

[Sample Input](#)

## responses

### status:

200 - OK

### representations

#### application/json

Example:

The response will contain a json object similar to the response of add device API.

[Sample response](#)

## https://<ipaddress>/emsam/nbi/device/suspend

### Methods

#### POST

suspendDevices ()

API to suspend a device.

### request

### representations

#### application/json

Example:

This API can be used to suspend a single or multiple devices.

A json with an array of device ipaddress should be provided as input.

[Sample input](#)

## responses

### status:

200 - OK

### representations

#### application/json

Example:

The response will contain a json object similar to the response of add device API.

[Sample response](#)

## https://<ipaddress>/emsam/nbi/device/resume

### Methods

#### POST

resumeDevices ()

API to resume management of a device.

### request

### representations

#### application/json

Example:

This API can be used to resume management of a single or multiple devices.  
A json with an array of device ipaddress should be provided as input.  
[Sample input](#)

#### responses

##### status:

200 - OK

##### [representations](#)

#### application/json

Example:  
The response will contain a json object similar to the response of add device API.  
[Sample response](#)

## https://<ipaddress>/emsam/nbi/nbidoc

### Methods

#### GET

getWadl ()

#### request

unspecified

#### responses

##### status:

200 - OK

##### [representations](#)

application/vnd.sun.wadl+xml
------------------------------

application/xml
-----------------

## https://<ipaddress>/emsam/nbi/phone

### Methods

#### GET

getPhone ()

#### request

##### [query params](#)

id	<a href="#">long</a>
----	----------------------

ipAddress	<a href="#">string</a>
-----------	------------------------

sysName	<a href="#">string</a>
---------	------------------------

### responses

**status:**

200 - OK

[representations](#)

application/json
------------------

application/xml
-----------------

## https://<ipaddress>/emsam/nbi/callFailure

API to get number of failed calls per cluster for each location.

### Methods

GET	getPoorCalls ()
-----	-----------------

API to get number of failed calls and total calls per cluster for each location for past 1 hour.

### request

unspecified

### responses

**status:**

200 - OK

[representations](#)

application/json
------------------

Example:

The output of this API is a JSON that will contain the overall result of the operation.

The response will contain result per cluster per location.

[Sample response](#)

## https://<ipaddress>/emsam/nbi/endpoint

### Methods

POST	getEndpoints ()
------	-----------------

### request

[representations](#)

application/json
------------------

**responses**

**status:**

200 - OK

**representations**

application/json
------------------

application/xml
-----------------

GET
-----

getEndpoint ()

**request**

**query params**

ipAddress	<a href="#">string</a>
-----------	------------------------

sysName	<a href="#">string</a>
---------	------------------------

id	<a href="#">long</a>
----	----------------------

**responses**

**status:**

200 - OK

**representations**

application/json
------------------

application/xml
-----------------

## https://<ipaddress>/emsam/nbi/troubleshoot/start

**Methods**

PUT
-----

startTroubleshooting ()

API to start a troubleshooting session.

**request**

**representations**

application/json
------------------

Example:

This API can be used to start a troubleshooting session.

A json with parameters as Meeting ID, source device ID and destination device ID should be provided as input.

[Sample input](#)

#### responses

##### status:

200 - OK

##### representations

#### application/json

Example:

The response will contain a json object with troubleshooting session ID, status message and status code.

[Sample response](#)

## https://<ipaddress>/emsam/nbi/troubleshoot/report/{tsSessionId}

### Methods

#### GET

getReport ()

API to get the report for a troubleshooting session.

#### request

##### template params

tsSessionId [int](#)

#### responses

##### status:

200 - OK

##### representations

#### application/json

Example:

The response will contain a json object with troubleshooting report containing session details, troubleshooting details, path topology and metrics for each hop in path topology.

[Sample response](#)

## https://<ipaddress>/emsam/nbi/troubleshoot/stop

### Methods

#### PUT

stop ()

API to stop a troubleshooting session.

#### request

##### representations

<b>application/json</b>	<p>Example:</p> <p>This API can be used to stop a troubleshooting session. A json with parameters as troubleshooting session ID should be provided as input.</p> <p><a href="#">Sample input</a></p>
-------------------------	--

#### responses

**status:**

200 - OK

[representations](#)

<b>application/json</b>	<p>Example:</p> <p>The response will contain a json object with troubleshooting session ID, status message and status code.</p> <p><a href="#">Sample response</a></p>
-------------------------	--

## https://<ipaddress>/emsam/nbi/troubleshoot/status/{tsSessionId}

#### Methods

<b>GET</b>	status ()
------------	-----------

API to get the status of a troubleshooting session.

#### request

[template params](#)

tsSessionId	<a href="#">int</a>
-------------	---------------------

#### responses

**status:**

200 - OK

[representations](#)

<b>application/json</b>	<p>Example:</p> <p>The response will contain a json object with troubleshooting session ID, current troubleshooting status, and status code.</p> <p><a href="#">Sample response</a></p>
-------------------------	---

## https://<ipaddress>/emsam/nbi/event

This NBI service handles various operations related to fault

\* Retrieval of events based on ipAddress and other various complex filtering criteria.

[Generic error codes and Messages](#)

[Device specific error codes and Messages](#)

## Methods

**POST** getEvents ()

This API will return all the alarms that match the filtering criteria.

**request**  
*representations*

**application/json** Example:  
The **filter** and **sort** fields are not mandatory.  
The output of the API will be limited to 500 records.  
This API will need to be called multiple times providing data for the **pagination** field to retrieve more data.  
[Sample input data](#)

**responses**  
**status:**  
200 - OK  
*representations*

**application/json** Example:  
[Sample response data](#)

## https://<ipaddress>/emsam/nbi/nbiLicense

This NBI service handles requests related to licensing. APIs available to get the license count, consumed license count and to get the license type.

[License API specific error codes and Messages](#)

## Methods

**GET** getLicenseDetails ()

API to retrieve the license type, license counts and consumed license counts.

**request**  
unspecified

**responses**  
**status:**  
200 - OK  
*representations*

**application/json** Example:  
The response will contain a json object giving license limit per device category.  
[Sample response](#)

## https://<ipaddress>/emsam/nbi/alarm

This NBI service handles various operations related to fault, operations like clear, acknowledge, annotate of alarms.

Also retrieval of alarms based on ipaddress and other various complex filtering criteria.

[Generic error codes and Messages](#)

[Fault specific error codes and Messages](#)

### Methods

<b>POST</b>	getAlarms ()
-------------	--------------

This API will return all the device that match the filtering criteria.

**request**  
[representations](#)

<b>application/json</b>	Example: The <b>filter</b> and <b>sort</b> fields are not mandatory. The output of the API will be limited to 500 records. This API will need to be called multiple times providing data for the <b>pagination</b> field to retrieve more data. <a href="#">Sample input data</a>
-------------------------	---

**responses**  
**status:**  
200 - OK  
[representations](#)

<b>application/json</b>	Example: <a href="#">Sample response data</a>
-------------------------	--

## https://<ipaddress>/emsam/nbi/alarm/operate

### Methods

<b>POST</b>	operateOnAlarm ()
-------------	-------------------

API to perform CLEAR/ACKNOWLEDGE/UNACKNOWLEDGE/ANNOTATE operations on alarms.

**request**  
[representations](#)

<b>application/json</b>	Example: This API takes JSON as input. JSON can contain multiple alarms ids with the operation to be performed on that. If the device should be discovered using a credential profile set the useProfile field to true in the input json, for this.
-------------------------	---

[Sample Input](#)

**responses**

**status:**

200 - OK

**representations**

**application/json**

Example:

The output of this API is a JSON that will contain the overall result of the operation, values can be

SUCCESS - if the operation on alarm was successful.

FAILURE - if the operation on alarm failed.

[Sample response](#)

## https://<ipaddress>/emsam/nbi/user

This Service provides API to perform user management operations.

[Generic error codes and Messages](#)

[User Management specific error codes and Messages](#)

**Methods**

**GET**

getUser ()

Returns the user if exists.

**request**

**query params**

**userName** [string](#)

**responses**

**status:**

200 - OK

**representations**

**application/json**

Example:

[Sample response](#)

**PUT**

addUsers ()

API for adding new user or updating existing user. API support one or more user addition/updation.

Different user role values are ['REPORTVIEWER','HELPDESK', 'OPERATOR', 'NETWORKADMIN','SYSTEMADMIN','SUPERADMIN']

NOTE : While adding any higher ROLE to PC through API, ensure adding all the LOWER ROLES explicitly.

For example, Make sure while adding user with role NETWORKADMIN explicitly provide below roles like REPORTVIEWER, HELPDESK and OPERATOR. API expect caller should take care of passing all applicable roles when adding user.

**request**  
*representations*

**\*/\*** Example\*: [Sample Input](#)

**responses**

**status:**

200 - OK

*representations*

<b>application/json</b>	<p>Example: The output of this API is a JSON that will contain the overall result of the operation, values can be SUCCESS - if user added/updated successfully FAILURE - if add/update has if failed</p> <p>The response will also contain 'results' JSON array which will contain more details of operation USER_OPERATION_SUCCESS - If user added/updated successfully. USER_OPERATION_FAILURE - In event of failure. If failed check errorCode to get more information. <a href="#">Sample response</a></p>
<b>DELETE</b>	deleteUser ()

API to delete a user.

**request**  
*query params*

**username** [string](#)

**responses**

**status:**

200 - OK

*representations*

<b>application/json</b>	<p>Example: The output of this API is a JSON that will contain the overall result of the operation, values can be SUCCESS - if user deleted successfully FAILURE - if delete user if failed</p> <p>The response will also contain 'results' JSON array which will contain more details of operation.</p>
-------------------------	--

	<p>USER_OPERATION_SUCCESS - If deleted successfully.          USER_OPERATION_FAILURE - In event of failure. If failed check          errorCode to get more information.  <a href="#">Sample response</a></p>
--	--

<b>POST</b>	getUsers ()
-------------	-------------

This API will return all the users that matches the input user names.

**request**  
[representations](#)

<b>*/*</b>	<p>Example:          The output of the API will be limited to 500 records.          This API will need to be called multiple times providing data for the <b>pagination</b> field to          retrieve more data.  <a href="#">Sample input data</a></p>
------------	--

**responses**  
**status:**  
 200 - OK  
[representations](#)

<b>application/json</b>	<p>Example:          response json will be similar to that of get user operation but with a list of          users.  <a href="#">Sample response data</a></p>
-------------------------	---

## https://<ipaddress>/emsam/nbi/user/password

### Methods

<b>POST</b>	changePassword ()
-------------	-------------------

API to change password for a user.  
 Password should match below mentioned password rules:

1. Password should be minimum 8 characters and Maximum 20 character.
2. Password must contain at least one of each - lowercase letter, uppercase letter, number, and special character.
3. No character in the password can be repeated more than three times consecutively.
4. Password cannot contain non-ASCII characters.
5. Password cannot be the same as the username, and cannot be the username reversed.
6. Password cannot or contain "cisco", "ocsic".

**request**  
[query params](#)

<b>username</b>	<a href="#">string</a>
-----------------	------------------------

newPassword [string](#)

### responses

**status:**

200 - OK

[representations](#)

**application/json**

Example:

The output of this API is a JSON that will contain the overall result of the operation, values can be

SUCCESS - if user deleted successfully

FAILURE - if delete user if failed

The response will also contain 'results' JSON array which will contain more details of operation

USER\_OPERATION\_SUCCESS - If password changed successfully.

USER\_OPERATION\_FAILURE - In event of failure. If failed check errorCode to get more information.

[Sample response](#)

## https://<ipaddress>/emsam/nbi/user/customer

### Methods

**POST**

assignCustomerForUser ()

This API to replace customer assigned for a given user. API support one or more user customer update.

**request**

[representations](#)

**application/json**

Example\* :

[Sample Input](#)

### responses

**status:**

200 - OK

[representations](#)

**application/json**

Example:

The output of this API is a JSON that will contain the overall result of the operation, values can be

SUCCESS - if user added/updated successfully

FAILURE - if add/update has if failed

The response will also contain 'results' JSON array which will contain more details of operation

USER\_OPERATION\_SUCCESS - If user added/updated successfully.  
USER\_OPERATION\_FAILURE - In event of failure. If failed check  
errorCode to get more information.  
[Sample response](#)

## https://<ipaddress>/emsam/nbi/credential

This service provides APIs to create, retrieve, update and delete credentials.

[Generic error codes and Messages](#)

[Device specific error codes and Messages](#)

### Methods

**POST** getCredentials ()

This API will return all the credential objects that match the filtering criteria.

#### request representations

**application/json** Example:  
The **filter** and **sort** fields are not mandatory.  
The output of the API will be limited to 500 records.  
This API will need to be called multiple times providing data for the  
**pagination** field to retrieve more data.

#### responses

**status:**

200 - OK

#### representations

**application/json** Example:  
The response json will be similar to that of getDevice operation but with  
a list of devices.  
[Sample response data](#)

**PUT** addCredential ()

API to add/edit credentials.

#### request representations

**application/json** Example:  
This API takes JSON as input. JSON can contain multiple credential  
objects (combination of credential profiles to add and modify).  
Credential profiles are uniquely identified by their profile names.  
Credential profile can be either a profile or a per-device credential.  
Per-device credentials apply only to a single device.  
Credential profiles can be applied to multiple devices based on the ip

address pattern or the device type.  
[Sample Input](#)

### responses

#### status:

200 - OK

#### representations

<b>application/json</b>	<p>Example:</p> <p>The output of this API is a JSON that will contain the overall result of the operation, values can be</p> <ul style="list-style-type: none"><li>SUCCESS - if the operation on all the credential profiles were successful.</li><li>FAILURE - if the operation on all the credential profiles failed.</li><li>PARTIALLY SUCCESSFUL - if the operation succeeded on some of the credential profiles.</li></ul> <p>The response will also contain result per credential profile which will contain the status as</p> <ul style="list-style-type: none"><li>SUCCESS - if the device was added successfully and a discovery job is created.</li><li>FAILURE - In the event of a failure to add a device the error code and message fields will also be populated.</li></ul> <p><a href="#">Sample response</a></p>
<b>GET</b>	getCredential ()

Returns the credential if exists.

### request

#### query params

<b>id</b>	<a href="#">long</a>
<b>name</b>	<a href="#">string</a>

### responses

#### status:

200 - OK

#### representations

<b>application/json</b>	<p>Example:</p> <p><a href="#">Sample response</a></p> <p>.</p>
<b>DELETE</b>	deleteCredential ()

API to delete a credential.

**request**  
*query params*

id	<a href="#">string</a>
----	------------------------

**responses**  
**status:**  
200 - OK  
*representations*

application/json
------------------

Example:  
The response will contain a json object similar to the response of add device API.  
[Sample response](#)

## https://<ipaddress>/emsam/nbi/diagnostictest

This service provides APIs to create diagnostics test.

[Generic error codes and Messages](#)

[Diagnostics test specific error codes and Messages](#)

### Methods

PUT	createDiagnosticTest ()
-----	-------------------------

**request**  
*representations*

application/json
------------------

**responses**  
**status:**  
200 - OK  
*representations*

application/json
------------------

application/xml
-----------------

POST	getDiagnosticTest ()
------	----------------------

**request**  
*representations*

application/json
------------------

**responses**  
**status:**  
200 - OK  
**representations**

application/json
------------------

application/xml
-----------------

## https://<ipaddress>/emsam/nbi/nbiDocumentation

### Methods

GET	getWadl ()
-----	------------

**request**  
unspecified

**responses**  
**status:**  
200 - OK  
**representations**

application/vnd.sun.wadl+xml
------------------------------

application/xml
-----------------