Cisco UCS Invicta API Guide

Version 1.6

August 22, 2014

Cisco Systems, Inc.
www.cisco.com

Cisco has more than 200 offices worldwide.
Addresses, phone numbers, and fax numbers
are listed on the Cisco website at
www.cisco.com/go/offices.

Text Part Number:

Cisco UCS Invicta Appliance Installation and Service Guide

CONTENTS

Preface

Organization

This guide includes the following sections:

Section	Title	Description
1	Overview	Explains the application programming interface (API) and all of its properties
2	API Methods	Provides the APIs and their properties

Conventions

This document uses the following conventions:

Convention	Indication
bold font	Commands and keywords and user-entered text appear in **bold** font.
italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
[ ]	Elements in square brackets are optional.
{x | y | z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
`courier` font	Terminal sessions and information the system displays appear in `courier` font.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Tip Means *the following information will help you solve a problem*. The tips information might not be troubleshooting or even an action, but could be useful information, similar to a Timesaver.

Caution Means *reader be careful*. In this situation, you might perform an action that could result in equipment damage or loss of data.

Timesaver Means *the described action saves time*. You can save time by performing the action described in the paragraph.

Warning **IMPORTANT SAFETY INSTRUCTIONS**

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.

SAVE THESE INSTRUCTIONS

Warning **Statements using this symbol are provided for additional information and to comply with regulatory and customer requirements.**

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation* at: http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html.

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation, as an RSS feed and deliver content directly to your desktop using a reader application. The RSS feeds are a free service.

Overview

API Structure

This document explains the application programming interface (API) and all of its properties that can be used to maintain and operate Cisco UCS Invicta Series Solid State Systems.

UCS Invicta Series Solid State Systems API was designed and implemented as RESTful Web Service. All API methods are accessible over HTTPS request using the URL that consists of the following parts:

- The domain: The IP of the Cisco UCS Invicta Series Solid State Systems
- The URI: `/restapi/<API_version>/<token>/<methodName>/<URI_reminder>`

A URI reminder is not always required but is essential to any methods that are requested over PUT or DELETE.

API Versions

The <API_version> in the URI (Example: 16) is important for consistent API support and compatibility. As the future enhancements and updates to the API methods are implemented, support for earlier versions is accessible. Each API method will preserve its signature as well as the response data structure under the version it was released, while all enhancements and changes is available under a new API version number.

The `<token>` URI element is included as a security measure to provide basic level of protection against unauthorized API use. See authenticate method section for all details.

JSON

Required data envelope on input (for POST and PUT request raw data) is JSON object notation labeled ‘data’.

The response object structure consists of `envelope_version`, `status`, `message` and `data`. Please note that the data portion could be empty if there is no data to return.

Here is a sample output of the raw JSON string following an API method call to find:

```
{"status":1,"message":"OK","data":{"unitName":"devMachine","apiVer-
sions":["15","16"]},"envelope_version":"15"}
```

cURL Examples

For cURL usage, please follow the examples below. Note that the <ip_address> and <token> entities should be replaced with the actual IP address and token, respectively.

- **GET**

```
curl -k -X GET --url https://<ip_address>/restapi/16/<token>/raidshealth
-H "Accept: application/json"
```

- **POST**

```
curl -X POST -d data='{"name":"lun1","vg":"ssd1","size":2,"strip-
ing":true}' -k --url https://<ip_address>/restapi/16/<token>/lun -H
"Accept: application/json"
```

- **PUT**

```
curl -X PUT -d data='{"size":3}' -k --url https://<ip_ad-
dress>/restapi/16/<token>/lun/lun1 -H "Accept: application/json"
```

- **DELETE**

```
curl -X DELETE -d data='{"name":"bond0:1"}' -k --url https://<ip_ad-
dress>/restapi/16/<token>/virtualinterface/0 -H "Accept: applica-
tion/json"
```

API Access/Permissions

A user account must be assigned to a role that has been granted API access permission. Permissions for individual API methods are dependent and directly related to a user's assigned roles and its associated permissions in the UI. For example, if a user is not assigned to a role that has permission to access the LUNs settings page in the UI, that user will also not have access to any of the LUN settings that are related to the LUN setting API methods. Similarly, if a user was assigned a role in the UI that has permission to add and edit LUNs s but does not have LUN delete permission, the same restriction will apply in the API.

Supported HTTP Methods

Version	GET /list	POST	PUT	DELETE
1.5	X	X	X	X
1.6	X	X	X	X

Supported Incoming Data Formats

Version	JSON	XML
1.5	X	
1.6	X	

API Methods

This section provides all of the available API methods. Data requirements, expected responses, dependencies, and any other pertinent information are all clearly defined in their respective calls. The information sent within a receivable envelope is handled accordingly to execute the request. Any data sent that is outside of the scope of the API method is ignored.

API Section Definitions

The API methods are documented using the sections cited in

Table 2-1 API Section Definitions

API Section Name	Description
Method Name	The method name is the exact syntax to be used for the API call.
Devices	The devices to which the API method applies.
Method	The method is the HTTP METHOD sent as part of the request to the device. The method can be POST, PUT, GET, or DELETE. • To create a resource on the server, use POST. • To retrieve a resource, use GET. • To change the state of a resource or to update it, use PUT. • To remove or delete a resource, use DELETE.
Synopsis	The synopsis provides a quick summary of the purpose of the method.
URL Example	The URL example is the model in which the API method is called. See the introduction for more details regarding the structure of the URL. Any PUT and DELETE methods require an extra parameter in the URI. This acts as the unique identifier (ID) of the object that needs to be modified. However, not all PUT or DELETE methods will pass the ID in the URI, as the ID may include an invalid character that would halt the method. Instead, a bogus ID (usually zero) is passed in the URI and the actual ID is sent as data. Keep in mind that the examples employ an artificial IP address. Substitute the example with the actual IP of the machine.

API Section Name	Description
Request/Response Data	The request and response data sections in each method description cover only the data portion of the returning envelope. The data types of all primitives must match according to the details of the method as described in the following pages. If the response data is of an array type, it is further explained (unless it is one-dimensional).
JSON Sample	A JSON sample of the data output is included in the documentation, if applicable.
Dependencies	The dependencies listed in each method are the devices or objects that are required to make the method work properly.
Notes	The notes are the additional comments that explain and provide clarification of different parts of the method.

FIND (Get)

Method name

find

Devices

Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method

GET

Synopsis

This method provides auto-discovery functionality. Every Cisco UCS Invicta Series Solid State System can be found by responding to this call.

Note Note: This is the only API method that does not require either version or token to be present in the URI.

URL example

```
https://10.1.39.127/restapi/find
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
unitId	String	Device serial number.
apiVersion*	Array	An array of strings
device	String	Information whether the device is Cisco UCS Invicta Scaling System or Cisco UCS Invicta Appliance.
boxSerial	String	Box serial number.
rrVersion	String	Currently installed RaceRunner software version.

JSON Sample

```
{"status":1,"message":"OK","data":{"apiVersion":[15,
16],"unitId":"WT-INV-QA-0000003","device":"INVICTA", "rrVersion":"Version 5.0.0 Release
004","boxSerial":"WT-INV-QA-SSRA-0000003"},"envelope_version":15}
```

Dependencies

n/a

Notes

- A token is not needed to use this API method call.
- Incoming data is not needed to perform this API method call.
- *apiVersions is returned as an array of strings.
- *unitId* and *boxSerial* can be used on Cisco UCS Invicta Scaling System for identification of different SSRs on the same Cisco UCS Invicta Scaling System.

AUTHENTICATE (POST)

Method name

authenticate

Devices

Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method

POST

Synopsis

This method returns a 32-bit token that can be used to execute many of the available API methods. Tokens last for up to an hour.

URL example

```
https://10.1.39.127/restapi/16/authenticate
```

Request data

NAME	TYPE	NOTES
username	String	Required
password	String	Required

JSON sample

```
'{"username":"jdoe", "password":"password"}'
```

Response data

NAME	TYPE	NOTES
token	String	Used in tandem to execute specific API calls.

JSON sample

```
"data":{"token":"1234567890xxxxxxxxxxxxxxxxxxxxx"}
```

Dependencies

n/a

Notes

- A token is not needed to use this API method call.
- Incoming data is needed to perform this API method call.
- Tokens can be reused as many times as needed until it expires.

USER (GET)

Method name user

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis Returns user information for the currently logged-in user.

URL example

```
https://10.1.39.127/restapi/16/<token>/user
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
`first_name`	String	User's first name as it is recorded in the system
`last_name`	String	User's last name as it is recorded in the system
`email_address`	String	User's email address as it is recorded in the system
`username`	String	User's login name

JSON sample

```
"data":{"first_name":"Joe","last_name":"Doe","email_address":"jdoe@domain.com",
"username", "joedoe"}
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.

USER (PUT)

Method name user

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method PUT

Synopsis Change the user information for the current user.

URL example

```
https://10.1.39.127/restapi/16/<token>/user
```

Request data

NAME	TYPE	NOTES
first_name	String	User's first name as recorded in the system
last_name	String	User's last name as recorded in the system
email_address	String	User's email address as recorded in the system
password	String	User's password

json sample

```
{"password":"xxxxx"}
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is needed to perform this API method call.

UNITINFO (GET)

Method name

unitinfo

Devices

Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method

GET

Synopsis

This method returns the model name, unit ID and name, total and available capacity of the unit, uptime, and IP address of the storage device.

URL example

```
https://10.1.39.127/restapi/16/<token>/unitinfo
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
modelName	String	Name of the device model.
unitId	String	Device serial number.
unitName	String	Host name.
totalCapacity	Integer	Total capacity in Bytes.
availableCapacity	Integer	Available capacity in Bytes.
systemUptime	String	System uptime information.
IPs	Array	An array of IP objects

IP Object

NAME	TYPE	NOTES
ipDevice	String	Name of the network interface.
ip	String	An IP assigned to this network interface.

JSON sample

```
"data":{"modelName":"INVICTA","unitId":"
WT-INV-PROD-0000001","unitName":"SSRA","totalCapacity":1187472557998,"availableCapacity":1
143492092887,"systemUptime":"4 days 03:24:15.46","IPs":[{“ipDevice”:”bond1”,
“ip”:”192.168.110.116”}]}
```

Dependencies

n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.

UNITNAME (GET)

Method name unitname

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis This method returns the unit name of the storage device.

URL example

```
https://10.1.39.127/restapi/16/<token>/unitname
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
unitName	String	Host name.

JSON sample

```
"data":{"unitName":"SSRA"}
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.

UNITNAME (PUT)

Method name unitname

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method PUT

Synopsis This method changes the unit name of the storage device.

URL example

```
https://10.1.39.127/restapi/16/<token>/unitname/0
```

Request data

NAME	TYPE	NOTES
unitName	String	New host name (required)

JSON sample

```
{"unitName":"HeadOne"}
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is needed to perform this API method call.
- The string `unitName` must be alphanumeric, may include dashes and periods, and cannot be longer than 15 characters.

Note Important: A zero (0) is included as part of the URI to conform to the PUT standards of the API application. Without the fulfilled parameter, the request will fail.

MODEL (GET)

Method name model

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method PUT

Task IDSynopsis This method returns the model of the device.

URL example

```
https://10.1.39.127/restapi/16/<token>/model
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
modelName	String	Name of the device model.

JSON sample

```
"data":{"modelName":"INVICTA"}
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.

RAIDHEALTH (GET)

Method name raidhealth

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis This method returns the health of all RAID units.

URL example

```
https://10.1.39.127/restapi/16/<token>/raidhealth
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
raids	Array	Explained below
nodes	Array	

Raids have array keys identical to the name of the actual RAID unit. Within those keys is the informational data about the RAID, with the following:

NAME	TYPE	NOTES
name	String	
status	Char	G = Good B = Bad D = Degraded R = Recovering
ssn	String	The SSN to which the raid belongs.

JSON sample

```
"data":{"nodes":{"SSN1":{"raids":[{"name":"rs0","status":"G"}]},"SSN2":{"raids":[{"name":"
rs0","status":"G"},{"name":"rs1","status":"G"},{"name":"rs2","status":"G"}]}
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.

SSN (GET)

Method name ssn

Devices Cisco UCS Invicta Scaling System

HTTP method GET

Synopsis Returns a list of SSNs with names and allocation status with the name of the volume group it was allocated to. If there is no allocation status, null is returned

URL example

```
https://10.1.39.127/restapi/16/<token>/ssn
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
	Array	An array of SSN objects

Each SSN is represented by an SSN object.

SSN Object

NAME	TYPE	NOTES
`name`	String	Name of the SSN
`allocation_status`	Bool	If SSN is allocated
`allocated_vg`	String||Null	Name of the VG to which the SSN is allocated

JSON sample

```
"data":[{"name":"SSN1","allocation_status":true,"allocated_vg":"vg1"}]
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.

- Incoming data is not needed to perform this API method call.

VOLUMEGROUP (GET)

Method name volumegroup

Devices Cisco UCS Invicta Scaling System

HTTP method GET

Synopsis Returns a list of all Volume Groups with name, size, available space, list of SSNs and status

URL example

```
https://10.1.39.127/restapi/16/<token>/volumegroup
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
	Array	Array of Volume Group objects

Volume Group Object

NAME	TYPE	NOTES
`name`	String	The name of the volume group
`size`	Float	The total size of the volume group in GB
`size_available`	Float	Total available space in volume group in GB
`ssns`	Array	Array of all bound SSNs
`status`	String	Volume group status

JSON sample

```
"data":[{"name":"vg1","size":100000,"size_available":80000,"ssns":["ssn1","ssn2"],"status"
:"good"}]
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.

VOLUMEGROUP (POST)

Method name volumegroup

Devices Cisco UCS Invicta Scaling System

HTTP method POST

Synopsis Creates a volume group.

URL example

```
https://10.1.39.127/restapi/16/<token>/volumegroup
```

Request data

NAME	TYPE	NOTES
vgName	String	Name of new volume group
ssn	Array	Array of SSNs to which new volume group is allocated

JSON sample

```
'{"vgname":"vg3", "ssn": ["ssn3", "ssn4"]}'
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is needed to perform this API method call.

VOLUMEGROUP (DELETE)

Method name volumegroup

Devices Cisco UCS Invicta Scaling System

HTTP method DELETE

Synopsis Deletes a volume group

URL example

```
https://10.1.39.127/restapi/16/<token>/volumegroup/<existing target volume group>
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.

ISCSIAVAILABLE (GET)

Method name iscsiavailable

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis Returns a list of all available IP addresses not yet assigned.

URL example

```
https://10.1.39.127/restapi/16/<token>/iscsiavailable
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
ip	Array	Array of IP Info Objects

JSON sample

```
"data": {"ip":["10.127.15.135","192.168.3.17"]}
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.

ISCSIAVAILABLE (PUT)

Method name iscsiavailable

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method PUT

Synopsis Add valid IP addresses to available and remove them from allowed

URL example

```
https://10.1.39.127/restapi/16/<token>/iscsiavailable/0
```

Request data

NAME	TYPE	NOTES
ip	String	IP address

JSON sample

```
{"ip": "10.127.15.135"}
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.

Note Important: A zero (0) is included as part of the URI to conform to the PUT standards on the API application. Without the fulfilled parameter, the request will fail.

ISCSIALLOWED (GET)

Method name iscsiallowed

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis Returns a list of all allowed IP addresses

URL example

```
https://10.1.39.127/restapi/16/<token>/iscsiallowed
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
ip	Array	Array of IP address

JSON sample

```
"data": {"ip" : ["10.127.15.207", "192.168.3.18"]}
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.

ISCSIALLOWED (PUT)

Method name iscsiallowed

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method PUT

Synopsis Add valid IP addresses to allowed and remove them from available

URL example

```
https://10.1.39.127/restapi/16/<token>/iscsiallowed/0
```

Request data

NAME	TYPE	NOTES
ip	String	IP address

JSON sample

```
{"ip":"10.127.15.162"}
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.

Note Important: A zero (0) is included as part of the URI to conform to the PUT standards on the API application. Without the fulfilled parameter, the request will fail.

LUN (GET)

Method name lun

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis This method returns the LUNS of the device.

URL example

```
https://10.1.39.127/restapi/16/<token>/lun
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
luns	Array	

LUN records have array keys consisting of the Volume Group and actual LUN name joined by ‘-‘ dash sign. Each record contains following LUN detail information:

NAME	TYPE	NOTES
name	String	LUN name
t10id	String	LUN T10ID string
lunid	String	LUN ID
size	Integer	LUN size
volumegroup	String	Volume group name this LUN belongs to
mirror	String	Name of the counterpart in mirroring if applicable
origin	String	Origin volume name or LUN which snapshot is associated with. Returns ‘-‘for non-snapshot volumes.
snap_pct	String	Shows current percentage of snapshot volume being used. Returns ‘-‘for non-snapshot volumes.

NAME	TYPE	NOTES
created	String	Shows when the snapshot was created. Returns '-' for non-snapshot volumes.
adminStatus	Integer	0 – Offline 1 – Online 2 - Degraded

JSON sample


```
"data":{"luns":{"VG1-L2":{"name":"L2","lunid":"1987cb76-L2","t10id":"
1987cb76-L2","size":10,"volumegroup":"VG1","mirror":"","origin":"-","
snap_pct":"-","created":"1409165442","adminStatus":"1"},"VG1-L3":{"na
me":"L3","lunid":"198790e0-L3","t10id":"198790e0-L3","size":20,"volum
egroup":"VG1","mirror":"","origin":"-","snap_pct":"-","created":"1409
164250","adminStatus":"1"}}}
```


Dependencies

n/a

Notes

- A token must be submitted in the URI to use this API method call.

LUN (POST)

Method name lun

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method POST

Synopsis This method creates a new LUN on the device.

URL example

```
https://10.1.39.127/restapi/16/<token>/lun
```

Request data

NAME	TYPE	NOTES
name	String	LUN name (required)
vg	String	Volume group name (required)
size	Integer	LUN size (required) Value in GB
striping	Boolean	Optional This property is ignored for Cisco UCS Invicta C3124SA Appliance

JSON sample

```
'{"name":"lun1", "vg":"vg1", "size":10}'
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The size should be specified in gigabytes (GB). Example: 2 = 2 GB.

LUN (PUT)

Method name lun

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method PUT

Synopsis This method increases the LUN size on the device.

URL example

```
https://10.1.39.127/restapi/16/<token>/lun/<name>
```

Request data

NAME	TYPE	NOTES
size	Integer	Value in GB Cannot be lower than previous value

JSON sample

```
{"size":1234}
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The <name>, submitted in the URI, is the unique name of the LUN that needs to be modified.
- The size (GB) cannot be lower than the previous value.

LUN (DELETE)

Method name lun

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method DELETE

Synopsis This method deletes a LUN from the device.

URL example

```
https://10.1.39.127/restapi/16/<token>/lun/<name>
```

Request data

NAME	TYPE	NOTES
n/a		

JSON sample

```
{"breakMode":"deleteTarget"}
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.
- During the LUN deletion process, GUI performance may be negatively affected until the process completes. The larger the LUN, the more time the process may take. We recommend deleting larger LUNs during off-peak hours.

LUNMIRROR (POST)

Method name lunmirror

Devices Cisco UCS Invicta Scaling System

HTTP method POST

Synopsis Create a mirror of an existing LUN.

URL example

```
https://10.1.39.127/restapi/16/<token>/lunmirror
```

Request data

NAME	TYPE	NOTES
lun	String	The LUN being mirrored
mirror	String	Name of the new LUN as a result of the mirror
vg	String	Name of available VG different than VG used by source LUN

JSON sample

```
'{"lun":"lun1", "mirror":"lun1mirror", "vg":"vg1"}'
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Depending on the size of the LUN, it may take some time to receive a response from this call.

LUNMIRROR (DELETE)

Method name lunmirror

Devices Cisco UCS Invicta Scaling System

HTTP method DELETE

Synopsis Break a LUN mirror

URL example

```
https://10.1.39.127/restapi/16/<token>/lunmirror/<lun id>
```

Request data

NAME	TYPE	NOTES
breakMode	string	Break mode: *keep* – breaks mirror and keeps both luns *deleteTarget* – breaks mirror and deletes target lun *deleteSource* – breaks mirror and deletes source lun

JSON sample

```
'{"breakMode":"deleteTarget"}'
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.
- The LUN name, not the LUN mirror name, is passed in this call.

TOTALSPACE (GET)

Method name totalspace

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis This method returns the total space available on the entire device.

URL example

```
https://10.1.39.127/restapi/16/<token>/totalspace
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
	Array	List of all volume groups
name	String	
freeSpace	Integer	Calculated in GBs

JSON sample

```
"data":[{"name":"vg1","freeSpace":"822"},{"name":"vg2","freeSpace":"818"
}]
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this API method call.

INITIATORGROUP (GET)

Method name initiatorgroup

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis This method returns all initiator groups with list of all initiators and assigned LUNs.

URL example

```
https://10.1.39.127/restapi/16/<token>/initiatorgroup
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
	Array	List of all initiator groups
`name`	String	Initiator group name
`initiators`	Array of strings	List of all initiators that belong to this particular initiator group
`luns`	Array of strings	List of all LUNs that are assigned to this particular initiator group

JSON sample

```
"data":[{"name":"IGsomename","initiators":[],"luns":["dev1:0"]},{"name":
"vmwaretest","initia-
tors":["iqn.1998-01.com.vmware:wt-lab-esx01-2229f4c5"],"luns":["dev:3","
dev1:0"]}]
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is not needed to perform this method call.

INITIATORGROUP (POST)

Method name initiatorgroup

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method POST

Synopsis This method adds an initiator group on the device.

URL example `https://10.1.39.127/restapi/16/<token>/initiatorgroup`

Request data

NAME	TYPE	NOTES
name	String	Required It is checked for uniqueness

JSON sample {“name”:”IG123”}

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- Incoming data is needed to perform this method call.

INITIATORGROUP (DELETE)

Method name initiatorgroup

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method DELETE

Synopsis This method deletes an initiator group on the device.

URL example `https://10.1.39.127/restapi/16/<token>/initiatorgroup/<group>`

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The `<group>` is the unique name of the initiator group that is removed from the device.

INITIATOR (POST)

Method name Initiator

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method POST

Synopsis This method creates an initiator for an initiator group on the device.

URL example `https://10.1.39.127/restapi/16/<token>/initiator`

Request data

NAME	TYPE	NOTES
`name`	String	Required Name of the initiator group
`initiator`	String	Required Checked for unique assignment – initiator can only be assigned to one group

JSON sample {“name”:”IG123”, “initiator”:”iqn.1998-01.com.vmware:wt-lab-esx01-2229f4c5”}

Response data

NAME	TYPE	NOTES
n/a		

Dependencies At least one initiator group must be available to use this method.

Notes

- A token must be submitted in the URI to use this API method call.
- The `name` is the unique name of the initiator group that the initiator is added to.
- There are 3 different formats supported for initiator string:
- Specific IQN format for iSCSI connection
- “Colon-separated” hex notation of WWN for FC connection
- “0X” hex notation of WWN for IB connection

INITIATOR (DELETE)

Method name initiator

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method DELETE

Synopsis This method deletes an initiator from an initiator group on the device.

URL example `https://10.1.39.127/restapi/16/<token>/initiator/<initiator_group>`

Request data

NAME	TYPE	NOTES
`name`	String	Required

JSON sample {“name”:”iqn.1998-01.com.vmware:wt-lab-esx01-2229f4c5”}

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The `name` is the initiator string specifying the initiator that is deleted from the `<initiator_group>`, submitted in the URI.

MAPLUN (POST)

Method name maplun

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method POST

Synopsis This method creates a LUN mapping on the initiator group level.

URL example `https://10.1.39.127/restapi/16/<token>/maplun`

Request data

NAME	TYPE	NOTES
`name`	String	Initiator group name – Required
`lunName`	String	LUN name – Required
`id`	Integer	Value between 0 and 255 Required

JSON sample {“name”:”IG123”, “lunName”:”lunt1”, “id”:3}

Response data

NAME	TYPE	NOTES
n/a		

Dependencies To utilize this method, a LUN and an initiator group must exist on the device.

Notes

- A token must be submitted in the URI to use this API method call.
- The `name` is the unique name of the initiator group.
- The `lunName` is the unique name of the LUN that is mapped to the initiator group.
- If there is no mapping with ID 0 (zero) present for the initiator group and *id* field in this request contains different number then 0 (zero), an error will be returned.

MAPLUN (DELETE)

Method name maplun

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method DELETE

Synopsis This method deletes a LUN mapping on the initiator group level.

URL example `https://10.1.39.127/restapi/16/<token>/maplun/<lun_name>`

Request data

NAME	TYPE	NOTES
`name`	String	Initiator group name – Required
`force_mapid0_delete`	Boolean	Optional If the LUN that will be unmapped has a MAP ID of 0, a warning is returned indicating MAP ID 0 is to be unmapped. Send this parameter to override warning.

JSON sample {“name”:”IG123”, “force_mapid0_delete“:true}

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The name is the unique name of the initiator group.
- The `lun_name` is the unique name of the LUN that is unmapped from the initiator group.

MAPLUNAUTO (POST)

Method name maplunauto

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method POST

Synopsis This method creates a LUN mapping on the initiator group level.

URL example `https://10.1.39.127/restapi/16/<token>/maplunauto`

Request data

NAME	TYPE	NOTES
`name`	String	Initiator group name – Required
`lunName`	String	LUN name – Required

JSON sample {“name”:”IG123”, “lunName”:”lunt1”}

Response data

NAME	TYPE	NOTES
n/a		

Dependencies To utilize this method, a LUN and initiator group must exist on the device.

Notes

- A token must be submitted in the URI to use this API method call.
- The `name` is the unique name of the initiator group.
- The `lunName` is the unique name of the LUN that is mapped to the initiator group.
- Next available mapping ID is being picked automatically.

WWN (GET)

Method name wwn

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis This method gets the Worldwide Name (WWN) of the device.

URL example

```
https://10.1.39.127/restapi/16/<token>/wwn
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
wwns	Array	An array of strings

JSON sample

```
"data":{"wwns":[{"0":"21:00:00:24:ff:39:fa:98"},{"1":"
21:00:00:24:ff:39:fa:99"}]}
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.

IQN (GET)

Method name iqn

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis This method gets the iSCSI Qualified Name (IQN) of the device.

URL example `https://10.1.39.127/restapi/16/<token>/iqn`

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
`iqn`	String	iSCSI Qualified Name

JSON sample

```
"data":{"iqn.2008-07.com.WHIPTAILtech:storage"}
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.

INTERFACES (GET)

Method name interfaces

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis This method gets all of the interfaces on the device.

URL example `https://10.1.39.127/restapi/16/<token>/interfaces`

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
`interfaces*`	Array	Array of Interfaces

`interfaces` has array keys identical to the name of the actual interface Each record contains following interface detail information:

NAME	TYPE	NOTES
`name`	String	Interface name
`mtu`	Integer	Maximum Transmission Unit
`status`	String	Interface status
`ip`	String	Interface IP
`mac`	String	Interface MAC address
`mask`	String	Interface network mask
`speed`	String	Only for physical devices
`interfaces`	Array	An array of strings Only for logical (bonds)
`maxSpeed`	String	Only for physical devices
`duplex`	String	Only for physical devices
`onboot`	String	”Yes” or “No”
`mode`	Integer	Empty is returned for Virtuals and Vlans.
`vlan`	Boolean	TRUE if this is vlan

virtualinterface	Boolean	TRUE if this is virtual interface
label	String	Applies only to un-bonded interfaces
mac	String	Mac address of un-bonded interface
status	String	Applies only to un-bonded interface
speed		Applies only to un-bonded interface N/A since there is no status data until the interface becomes bonded.
performance	String	Applies only to un-bonded interface N/A since there is no performance data until the interface becomes bonded.
type	String	Applies only to un-bonded interface The type of connection: TWISTED PAIR FIBRE DIRECT ATTACH COPPER

JSON sample


```
"data":{"interfaces":{"bond0":{"name":"bond0","status":"UP","ip":"10.10.
15.215","speed":"1000\/FULL","mtu":"1500","mask":"255.255.255.0","onBoot
":"yes","mode":"6","maxSpeed":null,"vlan":false,"virtualinterface":false
,"interfaces":["eth0"],"duplex":"FULL","mac":"00:50:56:B1:4E:FD"},"bond0
.1":{"name":"bond0.1","status":"UP","ip":"10.10.15.223","speed":"UNKNOWN
","mtu":"1500","mask":"255.255.255.0","onBoot":"yes","mode":"","maxSpeed
":null,"vlan":true,"virtualinterface":false,"interfaces":[],"duplex":nul
l,"mac":"00:50:56:B1:4E:FD"},"bond0:0":{"name":"bond0:0","status":"DOWN"
,"ip":"10.10.15.224","speed":"UNKNOWN","mtu":"1500","mask":"255.255.255.
0","onBoot":"yes","mode":"","maxSpeed":null,"vlan":false,"virtualinterfa
ce":true,"interfaces":[],"duplex":null,"mac":"00:50:56:B1:4E:FD"},"bond1
00":{"name":"bond100","status":"UP","ip":"10.90.90.2","speed":"1000\/FUL
L","mtu":"1500","mask":"255.255.255.0","onBoot":"yes","mode":"6","maxSpe
ed":null,"vlan":false,"virtualinterface":false,"inter-
faces":["eth1"],"duplex":"FULL","mac":"00:50:56:B1:02:5A"}}}
```


Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.

PHYSICALINTERFACES (GET)

Method name

physicalinterfaces

Devices

Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method

GET

Synopsis

This method gets all of the unbonded physical interfaces on the device.

URL example

```
https://10.1.39.127/restapi/16/<token>/physicalinterfaces
```

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
interfaces*	Array	Array of Interfaces

interfaces has array keys identical to the name of the actual interface Each record contains following interface detail information:

NAME	TYPE	NOTES
name	String	Interface name

JSON sample

```
"data":{"interfaces":[{"name":"eth5"},{"name":"eth6"}]]}
```

Dependencies

n/a

Notes

- A token must be submitted in the URI to use this API method call.

BOND (POST)

Method name bond

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method POST

Synopsis This method creates a new bond.

URL example

```
https://10.1.39.127/restapi/16/<token>/bond
```

Request data

NAME	TYPE	NOTES
ip	String	Bond IP
mask	String	Bond network mask
mtu	Integer	Maximum Transmission Unit
mode	Integer	Mode code – Required 0= Round-robin 1= Active Backup 4 = LACP 6 = ALB
onboot	Boolean	Initialize on boot – Required TRUE or FALSE

JSON sample

```
'{"ip":"10.10.10.120","mask":"255.255.255.0","mtu":1505,"mode":4,"onboot
":false}'
```

Response data

NAME	TYPE	NOTES
name	String	

JSON sample

```
"data":{"name":"bond1"}
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.

BOND (PUT)

Method name bond

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method PUT

Synopsis This method updates an existing bond.

URL example

```
https://10.1.39.127/restapi/16/<token>/bond/<bond_name>
```

Request data

NAME	TYPE	NOTES
ip	String	Bond IP –Required
mask	String	Bond network mask – Required
mtu	Integer	Maximum Transmission Unit – Required
mode	Integer	Mode code – Required 0= Round-robin 1= Active Backup 4 = LACP 6 = ALB
onboot	Boolean	Initialize on boot – Required TRUE or FALSE

JSON sample

```
'{"ip":"10.10.10.120","mask":"255.255.255.0","mtu":1505,"mode":4,"onboot
":false}'
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The bond_name in the URI is the unique name of the bond/device.

BOND (DELETE)

Method name bond

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method DELETE

Synopsis This method deletes an existing bond.

URL example https://10.1.39.127/restapi/16/<token>/bond/<bond_name>

Request data

NAME	TYPE	NOTES
n/a		

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The `bond_name` in the URI is the unique name of the bond/device.

TOGGLEBOND (PUT)

Method name togglebond

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method PUT

Synopsis This method enables/disables a bond device

URL example

```
https://10.1.39.127/restapi/16/<token>/togglebond/<bond_name>
```

Request data

NAME	TYPE	NOTES
`toggleState`	String	Action indicator -Required enable or disable

JSON sample

```
'{"toggleState":"enable"}'
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The `bond_name` in the URI is the unique name of the bond/device.

ENSLAVE (PUT)

Method name enslave

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method PUT

Synopsis This method adds or removes physical interface to an existing bond.

URL example

```
https://10.1.39.127/restapi/16/<token>/enslave/<bond_name>
```

Request data

NAME	TYPE	NOTES
`nicName`	String	Physical interface name e.g. eth2
`action`	String	add or remove

JSON sample

```
'{"nicName":"eth4", "action":"add"}'
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The `bond_name` in the URI is the unique name of the bond/device.

VLAN (POST)

Method name vlan

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method POST

Synopsis This method creates a new VLAN.

URL example

```
https://10.1.39.127/restapi/16/<token>/vlan
```

Request data

NAME	TYPE	NOTES
name	String	"Parent" interface name – Required
vlanID	Integer	VLAN ID – Required Value between 1 and 4096
ip	String	VLAN IP – Required
mask	String	VLAN network mask – Required
mtu	Integer	Maximum Transmission Unit – Required
onboot	Boolean	Initialize on boot – Required TRUE or FALSE
force	Boolean	If set to TRUE a check if IP is already in use is omitted – Optional TRUE or FALSE

JSON sample

```
'{"name":"bond0","vlanID":1,"ip":"10.10.15.221","mask":"255.255.255.0","
mtu":1500,"onboot":true,"force":true}'
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies To utilize this method, a bond must exist on the device.

Notes

- A token must be submitted in the URI to use this API method call.
- The name is the unique name of the bond/device.
- An error with status code 391 will be returned for any IP that can reply to a PING as an API will assume that the IP is already in use. The IP can still be used by sending the call again using the ‘force’ parameter.

VLAN (PUT)

Method name vlan

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method PUT

Synopsis This method updates an existing VLAN.

URL example

```
https://10.1.39.127/restapi/16/<token>/vlan/0
```

Request data

NAME	TYPE	NOTES
name	String	VLAN name – Required
ip	String	VLAN IP – Required
mask	String	VLAN network mask – Required
mtu	Integer	Maximum Transmission Unit – Required
onboot	Boolean	Initialize on boot – Required TRUE or FALSE
force	Boolean	If set to TRUE a check if IP is already in use is omitted – Optional TRUE or FALSE

JSON sample

```
'{"name":"bond0.1","ip":"10.10.15.222","mask":"255.255.255.0","mtu":1500
,"onboot":true,"force":true}'
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The name is the unique name of the bond/device and the vlanID joined by a period (Example: bond0.1234).

- An error with status code 391 will be returned for any IP that can reply to a PING as an API will assume that the IP is already in use. The IP can still be used by sending the call again using the ‘force’ parameter.

Note Important: A zero (0) is included as part of the URI to conform to the PUT standards of the API application. Without the fulfilled parameter, the request will fail.

VLAN (DELETE)

Method name vlan

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method DELETE

Synopsis This method deletes an existing VLAN.

URL example

```
https://10.1.39.127/restapi/16/<token>/vlan/0
```

Request data

NAME	TYPE	NOTES
name	String	

JSON sample

```
{"name":"bond3.1”}
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The name is the unique name of the bond/device and the vlanID joined by a period (Example: bond0.1234).

Note Important: A zero (0) is included as part of the URI to conform to the DELETE standards of the API application. Without the fulfilled parameter, the request will fail.

VIRTUALINTERFACE (POST)

Method name virtualinterface

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method POST

Synopsis This method creates a virtual interface.

URL example

```
https://10.1.39.127/restapi/16/<token>/virtualinterface
```

Request data

NAME	TYPE	NOTES
name	String	"Parent" interface name – Required
ip	String	Virtual interface IP – Required
mask	String	Virtual interface network mask – Required
mtu	Integer	Maximum Transmission Unit – Required
onboot	Boolean	Initialize on boot – Required TRUE or FALSE
force	Boolean	If set to TRUE a check if IP is already in use is omitted – Optional Required TRUE or FALSE

JSON sample

```
'{"name":"bond0","ip":"10.10.15.226","mask":"255.255.255.0","mtu":1500,"
onboot":true,"force":true}'
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies To utilize this method, a bond must exist on the device.

Notes

- A token must be submitted in the URI to use this API method call.
- The name is the unique name of the bond/device.
- An error with status code 391 will be returned for any IP that can reply to a PING as API will assume the IP is already in use. The IP can still be used by sending the call again using the ‘force’ parameter.

VIRTUALINTERFACE (PUT)

Method name virtualinterface

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method PUT

Synopsis This method updates a virtual interface.

URL example

```
https://10.1.39.127/restapi/16/<token>/virtualinterface/0
```

Request data

NAME	TYPE	NOTES
name	String	Virtual interface name – Required
ip	String	Virtual interface IP – Required
mask	String	Virtual interface network mask – Required
mtu	Integer	Maximum Transmission Unit – Required
onboot	Boolean	Initialize on boot – Required TRUE or FALSE
force	Boolean	If set to TRUE a check if IP is already in use is omitted – Optional TRUE or FALSE

JSON sample

```
'{"name":"bond0:0","ip":"10.10.15.227","mask":"255.255.255.0","mtu":1500
,"onboot":true,"force":true}'
```

Response data

NAME	TYPE	NOTES
n/a		

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The name is the unique name of the bond/device and the assigned ID joined by a colon (Example: bond1:1).

- An error with status code 391 will be returned for any IP that can reply to a PING as an API will assume that the IP is already in use. The IP can still be used by sending the call again using the ‘force’ parameter.

Note Important: A zero (0) is included as part of the URI to conform to the PUT standards of the API application. Without the fulfilled parameter, the request will fail.

VIRTUALINTERFACE (DELETE)

Method name virtualinterface

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method DELETE

Synopsis This method deletes a virtual interface.

URL example

```
https://10.1.39.127/restapi/16/<token>/virtualinterface/0
```

Request data

NAME	TYPE	NOTES
name	String	Virtual interface name – Required

JSON sample

```
{"name":"bond0:1”}
```

Response data

NAME	TYPE
n/a	

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.
- The name is the unique name of the bond/device and the assigned ID joined by a colon (Example: bond1:1).

Note Important: A zero (0) is included as part of the URI to conform to the DELETE standards of the API application. Without the fulfilled parameter, the request will fail.

PERFORMANCE – [TYPES] (GET)

Method name performance

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

Synopsis This method retrieves types of available performance logs.

URL example

```
https://10.1.39.127/restapi/16/<token>/performance/types
```

Request data

NAME	TYPE	NOTES
n/a		

URL request sample

```
curl -k -X GET --url 'https://10.10.15.181/restapi/16/<token>/perfor-
mance/types' -H "Accept: application/json"
```

Response data

NAME	TYPE	NOTES
	Array	An array of available types

JSON sample

```
"data":{"types":["bricks","luns","networkphysical","networkvirtual"]}
```

Dependencies n/a

Notes

- A token must be submitted in the URI to use this API method call.

PERFORMANCE – [RESOURCES] (GET)

Method name

performance

Devices

Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method

GET

Synopsis

This method retrieves available resources for the specified performance log type.

URL example

```
https://10.1.39.127/restapi/16/<token>/performance/resources/
type/<type>
```

Request data

NAME	TYPE	NOTES
type	String	Name of the performance log type.

URL request sample

```
curl -k -X GET --url 'https://10.10.15.181/restapi/16/<token>/perfor-
mance/resources/type/bricks' -H "Accept: application/json"
```

Response data

NAME	TYPE	NOTES
type	String	Name of performance log type
resources	Array	An array of available resources for the given type

JSON response

```
"data":{"type":"bricks","resources":["all","ssb1","ssb2"]}
```

Dependencies

The list of the available performance log types can be retrieved by API method PERFORMANCE – TYPES.

Notes

- A token must be submitted in the URI to use this API method call.

PERFORMANCE – [DATA] (GET)

Method name

performance

Devices

Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method

GET

Synopsis

This method retrieves data for the specified type and resource.

URL example

```
https://10.1.39.127/restapi/16/<token>/performance/data/type/
<type>/name/<name>/start/<start>/finish/<finish>/format/<format>
```

Request data

NAME	TYPE	NOTES
type	String	Name of the performance log type: bricks, luns, networkphysical, networkvirtual
name	String	Name of the resource For lun resource, the lun name given in format “<vgname>~<lunname>”
start	Datetime	Beginning of the time range given in format “yyyy-mm-dd HH:MM:ss”
finish	Datetime	Ending of the time range given in format “yyyy-mm-dd HH:MM:ss”
format	String	Optional parameter. Specifies return format: raw, json, zip

URL request sample

```
curl -k -X GET --url 'https://10.10.15.181/restapi/16/<token>/perfor-
mance/data/type/bricks/name/all/
start/2013-08-02%2012:33:00/finish/2013-08-02%2012:34:00/format/raw' -H
'Accept: application/json'
```

Response data

Response data varies depending on requested performance log type and returning format.

Type: bricks, luns

Format: raw

Response data:

NAME	TYPE	NOTES
	Array	An array of raw data Format: Unixdatetime, MBs, IOPS, Latency

JSON response sample


```
"data":"["1375461181,0.0,0.0,0.00","1375461186,0.0,0.0,0.00","1375461191
,0.0,0.0,0.00","1375461196,0.0,0.0,0.00","1375461201,0.0,0.0,0.00","1375
461206,0.0,0.0,0.00","1375461211,0.0,0.0,0.00","1375461216,0.0,0.0,0.00"
,"1375461221,0.0,0.0,0.00","1375461226,0.0,0.0,0.00","1375461231,0.0,0.0
,0.00","1375461236,0.0,0.0,0.00"]"
```


Type: networkphysical, networkvirtual

Format: raw

Response data:

NAME	TYPE	NOTES
	Array	An array of raw data. Format: Unixdatetime, rxpck, txpck, rxkb, txkb

JSON response sample


```
"data":"["1375288381,10.0,3.0,0.5,0.0","1375288386,11.0,3.0,0.5,0.0","13
75288391,7.0,3.0,0.0,0.0","1375288396,6.5,3.0,0.0,0.0","1375288401,7.0,3
.0,0.0,0.0","1375288406,5.5,3.0,0.0,0.0","1375288411,9.0,3.0,0.5,0.0","1
375288416,11.5,3.0,0.5,0.0","1375288421,6.5,3.0,0.0,0.0","1375288426,4.0
,3.0,0.0,0.0","1375288431,6.0,3.0,0.0,0.0","1375288436,8.5,3.0,0.0,0.0"]
"
```


Type: bricks, luns

Format: json

Response data:

NAME	TYPE	NOTES
	Array	DateTime in Unix Format

The structure of the array item:

NAME	TYPE	NOTES
mbs	Float	
iops	Float	
latency	Float	

JSON response sample

"data":"{"1375461181":{"mbs":0,"iops":0,"latency":0},"1375461186":{"mbs"
:0,"iops":0,"latency":0},"1375461191":{"mbs":0,"iops":0,"latency":0},"13
75461196":{"mbs":0,"iops":0,"latency":0},"1375461201":{"mbs":0,"iops":0,
"latency":0},"1375461206":{"mbs":0,"iops":0,"latency":0},"1375461211":{"
mbs":0,"iops":0,"latency":0},"1375461216":{"mbs":0,"iops":0,"latency":0}
,"1375461221":{"mbs":0,"iops":0,"latency":0},"1375461226":{"mbs":0,"iops
":0,"latency":0},"1375461231":{"mbs":0,"iops":0,"latency":0},"1375461236
":{"mbs":0,"iops":0,"latency":0}}"

Type: networkphysical, networkvirtual

Format: json

Response data:

NAME	TYPE	NOTES
	Array	DateTime in Unix Format

The structure of the array item:

NAME	TYPE	NOTES
rxpck	Float	
txpck	Float	
rxkb	Float	
txkb	Float	

JSON response sample

"data":"{"1375288381":{"rxpck":10,"txpck":3,"rxkb":0.5,"txkb":0},"137528
8386":{"rxpck":11,"txpck":3,"rxkb":0.5,"txkb":0},"1375288391":{"rxpck":7
,"txpck":3,"rxkb":0,"txkb":0},"1375288396":{"rxpck":6.5,"txpck":3,"rxkb"
:0,"txkb":0},"1375288401":{"rxpck":7,"txpck":3,"rxkb":0,"txkb":0},"13752
88406":{"rxpck":5.5,"txpck":3,"rxkb":0,"txkb":0},"1375288411":{"rxpck":9
,"txpck":3,"rxkb":0.5,"txkb":0},"1375288416":{"rxpck":11.5,"txpck":3,"rx
kb":0.5,"txkb":0},"1375288421":{"rxpck":6.5,"txpck":3,"rxkb":0,"txkb":0}
,"1375288426":{"rxpck":4,"txpck":3,"rxkb":0,"txkb":0},"1375288431":{"rxp
ck":6,"txpck":3,"rxkb":0,"txkb":0},"1375288436":{"rxpck":8.5,"txpck":3,"
rxkb":0,"txkb":0}}"

Type: any

Format: zip

Response data:

NAME	TYPE	NOTES
file	String	Filename

JSON response sample

"data":"{"file":"perf_bricks_all_20130802123300_20130802123400.log.gz"}"

To retrieve zip file **FILE** API method should be used.

Dependencies

The list of the available performance log types can be retrieved by API method PERFORMANCE – TYPES.

The available resources can be retrieved by API method PERFORMANCE – RESOURCES.

Notes

- A token must be submitted in the URI to use this API method call.
- The slash character must be substituted when the resource name ***lun*** is returned in the form of <*vgname*>/<*lunname*> (for example, the escaped slash character in the JSON format "vg1\/lun1"). In this instance the URL encoding will not work because while the URL is being processed internally, the '%2F' sequence is already translated back into '/' (slash) and the system tries to route the request to a non-existent location for response processing.

  For this reason '~' (tilde) was selected as a character to substitute for the '/' (slash) in the following example:

  ```
  'vg1/lun1' lun-resource name  "vg1~lun1"
  ```

FILE (GET)

Method name file

Devices Cisco UCS Invicta Scaling System, Cisco UCS Invicta Appliance

HTTP method GET

HTTP header Accept: application/octet-stream

Synopsis This method downloads the file.

URL example

```
https://10.1.39.127/restapi/16/<token>/file/<filename>
```

Request data

NAME	TYPE	NOTES
`filename`	String	File name from the PERFORMANCE – DATA when `zip` format was specified.

URL request sample

```
curl -k -O -X GET --url
'https://10.10.15.181/restapi/16/<token>/file/per-
f_bricks_all_20130802123300_20130802123400.log.gz' -H "Accept: applica-
tion/octet-stream"
```

Response data Streams the binary data.

Dependencies The list of the available performance log types can be retrieved by API method PERFORMANCE – DATA with specified data format as `zip`.

Notes

- A token must be submitted in the URI to use this API method call.
- In curl option –O should be specified in order to accept the filename same as source.

Status Codes

The Table 2-2 lists status codes.

Table 2-2 Status Codes

CODE	MESSAGE
1	OK.
2	OK – method deprecated.
100	Invalid data format.
101	HTTP method <type> not supported for `<function>` REST method.
102	Invalid credentials.
103	Not enough permissions.
104	Unexpected call.
105	Internal API Error.
106	Unsupported method for this device type.
107	Resource not found.
121	Validation for `<function>` has failed. Check `<primitive>` for correct data type/format.
320	Failed to delete Initiator Group.
321	Invalid Initiator Group ID.
322	Failed to create Initiator Group.
323	Initiator Group ID already exists.
324	Initiator is invalid.
325	Failed to add Initiator.
326	Failed to remove Initiator.
327	LUNS or NFS Volumes present in Volume Group. Failed to delete Volume Group.
328	Volume group does not exist. Failed to delete volume group.
329	Failed to delete Volume Group.
330	Partner has a lock. Failed to create volume group.
331	Failed to create Volume Group.
332	Failed to rename volume group.
333	Invalid Volume Group. Failed to create LUN.
334	Invalid Device Group. Failed to create LUN.
335	This LUN cannot be striped. Failed to create LUN.
336	The LUN name is not available. Failed to create LUN.
337	LUN does not exist. Failed to delete LUN.

Table 2-2 Status Codes (continued)

338	Amount must be greater than zero. Failed to update LUN.
339	Failed to update LUN.
340	Initiator Group does not exist.
341	Invalid Volume Group. Failed to mirror LUN.
342	Failed to mirror LUN.
343	LUN name is not available. Failed to mirror LUN.
344	Failed to create LUN mapping. The specific mapping you are trying to create may already exist.
345	The LUN name provided does not exist.
346	The ID provided is invalid.
347	The initiator group provided does not contain any initiators.
348	Failed to remove LUN Mirror
349	The LUN you are attempting to unmap is not mapped to the group you have provided.
350	Unable to get next free mapID.
351	Unreachable IP. Could not change partner IP.
352	Could not change partner IP.
353	IP address not available. Failed to create Bond.
354	Failed to create Bond.
355	Invalid Bond. Failed to delete Bond.
356	Failed to delete Bond.
357	Bond could not be found. Failed to update Bond.
358	Failed to update bond.
359	Invalid bond. Failed to create VLAN.
360	Failed to create VLAN.
361	VLANID is not available. Failed to create VLAN.
362	Unable to perform the enslaving action attempted.
363	Request does not match expect usage. Usage: `createBondedEth <bond> <eth>.`
364	Target Ethernet interface is being used. Confirm interface is not in use before trying again
365	The Ethernet interface provided does not exist.
366	The bond interface provided does not exist
367	Enslave action unrecognized. Expected actions `are < add | remove >.`
368	Portal is already allowed.
369	Portal does not exist.
370	IP address not available. Failed to update bond.
371	Parameter is invalid.
372	Unable to perform the bond action requested. Verify that bond exists in a DELETE or PUT request, and does not exist in a POST.
373	Failed to update user.

Table 2-2 Status Codes (continued)

375	Device does not exist.
376	LUN does not exist.
377	There is not enough space on the destination Volume Group.
378	Cannot connect to peer. Please try again later.
379	LUN mirror break failed: This LUN does not contain a mirror to break.
380	Unable to find serial number.
381	FIND failed. Unable to get `apiVersion`.
382	FIND failed. Unable to get `unitId.`
383	FIND failed. Unable to get `device.`
384	FIND failed. Unable to get `boxSerial.`
385	IP address not available. Failed to create Virtual Interface.
386	LUN is mirrored. Failed to update LUN.
387	IP is already in use. Failed to update VLAN
388	An error has occurred. Failed to perform action on VLAN.
389	IP is already in use. Failed to add VLAN.
390	Map ID 0 is about to be removed. Failure to remap a LUN 0 will result in initiator issues. Please use force_mapid0_delete parameter to force unmapping of this LUN.