Avocent® ACS800/8000 Advanced Console System

Application Programming Interface (API) User Guide

The information contained in this document is subject to change without notice and may not be suitable for all applications. While every precaution has been taken to ensure the accuracy and completeness of this document, Vertiv assumes no responsibility and disclaims all liability for damages resulting from use of this information or for any errors or omissions.

Refer to local regulations and building codes relating to the application, installation, and operation of this product. The consulting engineer, installer, and/or end user is responsible for compliance with all applicable laws and regulations related to the application, installation, and operation of this product.

The products covered by this instruction manual are manufactured and/or sold by Vertiv. This document is the property of Vertiv and contains confidential and proprietary information owned by Vertiv. Any copying, use, or disclosure of it without the written permission of Vertiv is strictly prohibited.

Names of companies and products are trademarks or registered trademarks of the respective companies. Any questions regarding usage of trademark names should be directed to the original manufacturer.

Technical Support Site

If you encounter any installation or operational issues with your product, check the pertinent section of this manual to see if the issue can be resolved by following outlined procedures.

Visit https://www.vertiv.com/en-us/support/ for additional assistance.

Proprietary and Confidential ©2025 Vertiv Group Corp.

TABLE OF CONTENTS

1 Overview **1**
1.1 Base URL 1
1.2 Methods 1
1.3 Body 1
1.4 Query Parameters 2
1.4.1 Fields 2
1.4.2 Filtering 2
1.4.3 Ranges 3
1.4.4 Format 3
1.5 Response Codes 3
1.5.1 Error information 4
1.5.2 Ignored keys 4
1.6 Authentication 5
1.6.1 JSON Web Token (JWT) 5
1.6.2 Basic authentication 5
1.7 Document Conventions 5
1.7.1 Abbreviated URLs 5
1.7.2 Examples 5
2 API Resources and Methods **7**
2.1 Sessions 20
2.1.1 /sessions/login 20
2.1.2 /sessions/logout 22
2.1.3 /sessions/refresh 23
2.1.4 /sessions[/<ID>] 24
2.2 System 26
2.2.1 /system/info 26
2.2.2 /system/reboot 28
2.2.3 /system/shutdown 29
2.2.4 /system/factoryDefault 30
2.2.5 /system/firmware 31
2.2.6 /system/firmware/version 33
2.2.7 /system/firmware/download 34
2.2.8 /system/firmware/install 36
2.2.9 /system/firmware/downloaded 38
2.2.10 /system/config 39
2.2.11 /system/config/save 40
2.2.12 /system/config/restore 43
2.2.13 /system/integrity/generate 45

2.2.14 /system/integrity/verify 46
2.2.15 /system/certificate/generate 47
2.2.16 /system/certificate/download 49
2.2.17 /system/certificate 51
2.2.18 /system/certificate/upload 53
2.2.19 /system/certificate/apply 55
2.2.20 /system/dateAndTime 56
2.2.21 /system/dateAndTime/testNtp 58
2.2.22 /system/dateAndTime/timezones 59
2.2.23 /system/dateAndTime/timezone/custom 60
2.2.24 /system/general 62
2.2.25 /system/bootConfig 64
2.2.26 /system/usage/memory 66
2.2.27 /system/usage/flash 68
2.3 Security Profile 70
2.3.1 /security 70
2.3.2 /security/clearDSView 75
2.4 Network 76
2.4.1 /network/settings 76
2.4.2 /network/devices[/<INTERFACE>] 79
2.4.3 /network/hosts[/<IPADDRESS>] 81
2.4.4 /network/staticRoutes/<TABLE>[/<ID>] 84
2.4.5 /network/firewall/<TABLE>[/<CHAIN>] 86
2.4.6 /network/firewall/<TABLE>/<CHAIN>/rules[/<INDEX>] 88
2.4.7 /network/firewall/<TABLE>/<CHAIN>/rules/<INDEX>/move 92
2.4.8 /network/ipsec/connections[/<NAME>] 93
2.4.9 /network/ipsec/connections/<NAME>/connect 97
2.4.10 /network/ipsec/connections/<NAME>/disconnect 98
2.4.11 /network/ipsec/connections/<NAME>/ping 99
2.4.12 /network/ipsec/connections/<NAME>/diagnostics 101
2.4.13 /network/ipsec/certificates 103
2.4.14 /network/ipsec/certificates/download 104
2.4.15 /network/gre/tunnels[/<NAME>] 106
2.4.16 /network/gre/tunnels/<NAME>/connect 109
2.4.17 /network/gre/tunnels/<NAME>/disconnect 110
2.4.18 /network/gre/tunnels/<NAME>/ping 111
2.4.19 /network/gre/tunnels/<NAME>/diagnostics 113
2.4.20 /network/snmp[/<ID>] 114
2.4.21 /network/snmp/system 117
2.4.22 /network/dhcpServerSettings 118

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.23 /network/dhcpServerAddresses[/<IPADDRESS>] 120
2.4.24 /network/certificates/settings 123
2.4.25 /network/certificates/status 125
2.4.26 /network/certificates/root 127
2.4.27 /network/certificates/root/<NAME> 128
2.4.28 /network/certificates/root/download 130
2.4.29 /network/certificates/root/upload 132
2.5 Serial Ports 133
2.5.1 /serialPorts[/<PORT>] 133
2.5.2 /serialPorts/<PORT>/enable 138
2.5.3 /serialPorts/<PORT>/disable 139
2.5.4 /serialPorts/<PORT>/clone 140
2.5.5 /serialPorts/<PORT>/resetToFactory 141
2.5.6 /serialPorts/<PORT>/discoverNow 142
2.5.7 /serialPorts/<PORT>/alerts 143
2.5.8 /serialPorts/<PORT>/alerts/<ID> 145
2.5.9 /serialPorts/<PORT>/alerts/clear 146
2.5.10 /serialPorts/<PORT>/alerts/deleteAny 147
2.5.11 /serialPorts/<PORT>/power 148
2.5.12 /serialPorts/<PORT>/power/<ID> 151
2.6 Auxiliary Ports 152
2.6.1 /auxPorts[/<NAME>] 152
2.7 Modems 155
2.7.1 /modems[/<NAME>] 155
2.7.2 /modems/<NAME>/signalCheck 160
2.7.3 /modems/<NAME>/registration 161
2.8 CAS Profile 162
2.8.1 /casProfile 162
2.8.2 /casProfile/probeStrings[/<INDEX>] 164
2.8.3 /casProfile/matchStrings[/<INDEX>] 166
2.8.4 /casProfile/commands[/<NAME>] 168
2.8.5 /casProfile/autoAnswer[/<INDEX>] 171
2.8.6 /casProfile/poolOfPorts[/<NAME>] 173
2.9 Dial-In Profile 176
2.9.1 /dialinProfile 176
2.9.2 /dialinProfile/callbackUsers[/<NAME>] 178
2.9.3 /dialinProfile/pppOtpUsers[/<NAME>] 180
2.9.4 /dialinProfile/callerId[/<INDEX>] 182
2.9.5 /dialinProfile/chapSecrets[/<NAME>] 184
2.10 Pluggable Devices 186

2.10.1 /pluggableDevices[/<NAME>] 186
2.10.2 /pluggableDevices/<NAME>/setConsole 188
2.10.3 /pluggableDevices/<NAME>/eject 189
2.10.4 /pluggableDevices/<NAME>/delete 190
2.11 Authentication 191
2.11.1 /authentication 191
2.11.2 /authentication/dsview 193
2.11.3 /authentication/duo 194
2.11.4 /authentication/kerberos 195
2.11.5 /authentication/ldap 196
2.11.6 /authentication/radius 198
2.11.7 /authentication/tacacs 200
2.12 Users 202
2.12.1 /users[/<NAME>] 202
2.12.2 /users/<NAME>/unlock 205
2.12.3 /users/<NAME>/disable 206
2.12.4 /users/<NAME>/enable 207
2.12.5 /users/passwordRules 208
2.13 Groups 210
2.13.1 /groups[/<NAME>] 210
2.14 Access Rights 213
2.14.1 /accessRights/users/<NAME>/serialPorts[/<PORT>] 213
2.14.2 /accessRights/users/<NAME>/appliance 216
2.14.3 /accessRights/users/<NAME>/pdus[/<ID>] 218
2.14.4 /accessRights/users/<NAME>/outlets[/<ID>] 220
2.14.5 /accessRights/users/<NAME>/ups[/<ID>] 222
2.14.6 /accessRights/users/<NAME>/upsOutletGroups[/<ID>] 224
2.14.7 /accessRights/groups/<NAME>/serialPorts[/<PORT>] 226
2.14.8 /accessRights/groups/<NAME>/appliance 226
2.14.9 /accessRights/groups/<NAME>/pdus[/<ID>] 226
2.14.10 /accessRights/groups/<NAME>/outlets[/<ID>] 226
2.14.11 /accessRights/groups/<NAME>/ups[/<ID>] 226
2.14.12 /accessRights/groups/<NAME>/upsOutletGroups[/<ID>] 226
2.14.13 /accessRights/dsview 227
2.15 Events and Logs 228
2.15.1 /events[/<ID>] 228
2.15.2 /events/syslog 230
2.15.3 /events/snmp 232
2.15.4 /events/sms 234
2.15.5 /events/email 235

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.15.6 /events/dsview 237
2.15.7 /events/trapForward[/<INDEX>] 239
2.15.8 /events/trapForward/settings 241
2.15.9 /events/dataBuffering 242
2.15.10 /events/applianceLogging 244
2.15.11 /events/test 246
2.16 Power Management 247
2.16.1 /power/pdus[/<NAME>] 247
2.16.2 /power/pdus/<NAME>/off 252
2.16.3 /power/pdus/<NAME>/on 253
2.16.4 /power/pdus/<NAME>/cycle 254
2.16.5 /power/pdus/<NAME>/rename 255
2.16.6 /power/pdus/<NAME>/resetValues 256
2.16.7 /power/pdus/<NAME>/reboot 258
2.16.8 /power/pdus/<NAME>/refresh 259
2.16.9 /power/pdus/<NAME>/factoryDefaults 260
2.16.10 /power/pdus/<NAME>/firmwareDownload 261
2.16.11 /power/pdus/<NAME>/firmwareInstall 263
2.16.12 /power/pdus/<NAME>/outlets[/<INDEX>] 264
2.16.13 /power/pdus/<NAME>/outlets/<INDEX>/on 267
2.16.14 /power/pdus/<NAME>/outlets/<INDEX>/off 268
2.16.15 /power/pdus/<NAME>/outlets/<INDEX>/cycle 269
2.16.16 /power/pdus/<NAME>/outlets/<INDEX>/lock 270
2.16.17 /power/pdus/<NAME>/outlets/<INDEX>/unlock 271
2.16.18 /power/pdus/<NAME>/outlets/<INDEX>/resetValues 272
2.16.19 /power/pdus/<NAME>/banks[/<INDEX>] 273
2.16.20 /power/pdus/<NAME>/banks/<INDEX>/resetValues 276
2.16.21 /power/pdus/<NAME>/phases[/<INDEX>] 277
2.16.22 /power/pdus/<NAME>/phases/<INDEX>/resetValues 280
2.16.23 /power/pdus/<NAME>/inlets[/<INDEX>] 281
2.16.24 /power/pdus/<NAME>/sensors[/<INDEX>] 283
2.16.25 /power/pdus/<NAME>/sensors/<INDEX>/resetValues 285
2.16.26 /power/ups[/<NAME>] 286
2.16.27 /power/ups/<NAME>/rename 294
2.16.28 /power/ups/<NAME>/restoreName 295
2.16.29 /power/ups/<NAME>/outputOn 296
2.16.30 /power/ups/<NAME>/outputOff 297
2.16.31 /power/ups/<NAME>/outputCycle 298
2.16.32 /power/ups/<NAME>/testBattery 299
2.16.33 /power/ups/<NAME>/silenceAlarm 300

2.16.34 /power/ups/<NAME>/resetPowerStats 301
2.16.35 /power/ups/<NAME>/outletGroups[/<INDEX>] 302
2.16.36 /power/ups/<NAME>/powerModules 304
2.16.37 /power/ups/<NAME>/batteryModules 306
2.16.38 /power/login 308
2.16.39 /power/outletGroups[/<NAME>] 310
2.16.40 /power/outletGroups/<NAME>/on 312
2.16.41 /power/outletGroups/<NAME>/off 313
2.16.42 /power/outletGroups/<NAME>/cycle 314
2.16.43 /power/outletGroups/<NAME>/outlets 315
2.16.44 /power/outletGroups/<NAME>/outlets/<ID> 317
2.16.45 /power/networkPdus[/<IPADDRESS>] 318
2.16.46 /power/networkUps[/<IPADDRESS>] 321
2.17 Sensors 324
2.17.1 /sensors/internal 324
2.17.2 /sensors/1Wire[/<ADDRESS>] 326
2.17.3 /sensors/1Wire/refresh 329
2.17.4 /sensors/digitalIn[/<INDEX>] 330
2.17.5 /sensors/pdu 332
2.17.6 sensors/pdu/<NAME>/reset 334
2.18 Digital Out 335
2.18.1 /digitalOut[/<INDEX>] 335
2.19 Monitoring 337
2.19.1 /monitoring/network/devices[/<INTERFACE>] 337
2.19.2 /monitoring/network/routingTables/ipv4 339
2.19.3 /monitoring/network/routingTables/ipv6 341
2.19.4 /monitoring/serialPorts[/<PORT>] 343
2.19.5 /monitoring/scheduledTasks[/<NAME>] 346
2.19.6 /monitoring/scheduledTasks/<NAME>/runNow 349
2.19.7 /monitoring/ipsec[/<NAME>] 350
2.19.8 /monitoring/autoDiscovery[/<PORT>] 352
2.19.9 /monitoring/autoDiscovery/<PORT>/discoverNow 354
2.19.10 /monitoring/fipsMode 355
2.19.11 /monitoring/callerIdLog 356
2.19.12 /monitoring/callerIdLog/clearLog 358
2.19.13 /monitoring/modemPppdLog 359
2.19.14 /monitoring/modemPppdLog/clearLog 360
2.19.15 /monitoring/zeroTouchLog 361
2.19.16 /monitoring/zeroTouchLog/clearLog 362
2.20 Access 363

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.20.1 /access/serialPorts[/<PORT>] 363
2.20.2 /access/serialPorts/<PORT>/on 365
2.20.3 /access/serialPorts/<PORT>/off 366
2.20.4 /access/serialPorts/<PORT>/cycle 367
2.21 Miscellaneous 368
2.21.1 /changePassword 368
2.21.2 /resources 370
Appendices 379
Appendix A: cURL 379
Appendix B: Python 380
Appendix C: Helper Script 381
Appendix D: Certificate Verification 383

This page intentionally left blank

Proprietary and Confidential ©2025 Vertiv Group Corp.

1 Overview

This document defines the native RESTful Application Programming Interface (API) for the Vertiv™ Avocent® ACS800/8000 advanced console system. This document is up to date with the 2.30 release of the Avocent ACS8000 console system firmware.

NOTE: Unless otherwise noted, all references to Vertiv™ management software in this document refer to the Vertiv™ Avocent® DSView™ Solution or Vertiv™ Avocent® DSView 4.5 management software.

1.1 Base URL

The base URL format is as follows:

https://<IP_ADDRESS>:<PORT_NUMBER>/api/v1/

The HTTPS port number is 48048 by default. HTTP may also be used with the default port number, 8080; however, this is completely insecure. By default, HTTP access is disabled and only HTTPS access is enabled. Both HTTP and HTTPS access can be enabled or disabled by the administrator and their respective port numbers changed. Changes to these settings will restart the RESTful API server and disconnect any existing RESTful sessions.

1.2 Methods

The following basic HTTP methods are supported.

Table 3.1 Supported HTTP Methods

Method	Description
POST	Used to create a new resource (specified in the JSON body) underneath the resource specified by the URL. The created resource is returned in the body or enough information is returned to find the new resource (an ID or URL). Also used to initiate actions.
GET	Requests a representation of the specified resource. This has no other effects other than reading and returning the data.
PUT	Modifies an existing resource (specified in the URL) with the data present in the JSON body. Only the items present in the body are modified and the rest are left unchanged. The response code is typically 204 (Status No Content) with no content in the response body unless otherwise requested by a specific parameter.
PATCH	Similar to the PUT method but typically intended only for modifying a portion of the specified resource. PUT in this API is also allowed to only modify a portion of a resource, so PATCH is included just for those applications that already use it.
DELETE	Deletes the resource specified by the URL.

1.3 Body

The body for GET/POST/PUT/PATCH requests uses JSON syntax. This means the "Content-Type" and "Accept" headers should be set to application/json. All parameters and values are case sensitive. Parameters that are string types must have their value enclosed in quotes even if the content is numerical, as is the case with some of the parameters that have a drop-down menu in the web User Interface (UI).

NOTE: All parameters are string type unless otherwise specified to be an integer, array or another type.

Proprietary and Confidential ©2025 Vertiv Group Corp.

1.4 Query Parameters

1.4.1 Fields

The fields query parameter is supported for many resources to enable you to limit the returned fields. The following example returns all the serial ports with only the speed and pinout fields of each:

```
GET /serialPorts?fields=pinout,speed
```

Sub-fields, such as speed which is under the physical portion of a serial port, are unique and can be specified without any reference to the parent (physical in this example).

1.4.2 Filtering

Filtering can be performed on certain resources that return an array of items by specifiying attributes of that resource that should be matched for each item.

Example 1:

```
GET /serialPorts?status=enabled&speed=9600
```

This request returns an array of serial ports that are enabled and set to a speed of 9600.

NOTE: The ampersand (&) is a standard URL delimiter and not meant to imply a logical "and" operation. A logical "and" is implicit because all specified filters must match for the element to be included in the returned resource list.

Example 2:

```
GET /serialPorts?speed=1200, 2400, 4800
```

For any single parameter, multiple acceptable values can be specified and separated by commas. This request returns an array of serial ports that have port speeds of 1200, 2400 or 4800.

Example 3:

```
GET /serialPorts?port=1,3-5,7
```

Ranges for integer fields can be specified using a hyphen (-) and can be combined with the comma separated list. This request returns an array of serial ports consisting of ports 1, 3, 4, 5 and 7.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Example 4:

```
GET /serialPorts?profile=~cas
```

A value prefixed with "~" inverts the condition. This request returns an array of serial ports with the profile NOT set to cas.

Example 5:

NOTE: When using both fields and filtering queries, the fields must include those that are used for filtering. For example:

```
GET /events?fields=syslog\&syslog=enabled
```

1.4.3 Ranges

Ranges are supported for certain numeric parameters when filtering and use the following format:

```
GET /serialPorts?port=1,3-5,7
```

This returns only the items specified.

1.4.4 Format

Format is supported for a few parameters that may return text, including escaped special characters, for which the user may prefer to have base64 encoded. The default format (text) is a standard text encoding with special characters escaped, but base64 can be specified as follows:

```
GET /system/certificate?format=base64
```

1.5 Response Codes

The API utilizes standard HTTP response codes where appropriate. The following table lists the supported response codes and typical usage.

Table 3.2 Response Code Descriptions

Response Code	Meaning	Description
200 OK	Success	Returned for a successful request, which may include a JSON body with results.
201 Created	Created	Returned for a successful request that has resulted in the creation of a new resource.
204 No Content	Success	Returned for a successful request but does NOT include a JSON body with results.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Table 3.2 Response Code Descriptions (continued)

Response Code	Meaning	Description
400 Bad Request	Failure	Returned on a failed request and includes a JSON encoded list of errors for one or more of the problematic parameters. May also indicate other system errors.
401 Unauthorized	Authorization Failure	Returned for a request without the proper authentication.
404 Not Found	API not active	Returned for a request where the resource is not found, typically including a JSON-encoded error structure with more error detail.

1.5.1 Error information

In addition to returning a failing response code, error information is returned in the response body to provide more detail. This response is in JSON format as follows:.

```
{
    "error": {
        "code": "AE003",
        "message": "invalid parameter",
        "detail": "bob"
    }
}
```

NOTE: Not all error information responses include the detail field.

1.5.2 Ignored keys

If unknown keys are sent as part of the JSON body of a PUT or PATCH request, they are ignored by the API. This is part of the RESTful way of supporting different devices and different versions of API implementation. If a device doesn't understand or support something, it is permitted to ignore it.

Rather than ignoring spelling errors and leaving the caller to wonder why a parameter wasn't set, the API returns a response body with the "200" response code that contains a successful response message and a list of keys that were ignored. This permits the caller to look for this information, if desired.

```
{
    "status": "success",
    "ignoredKeys": [
        "datea",
        "timee"
    ]
}
```

If a parent key is ignored, all children below it are not processed nor listed as ignored keys. Even if all keys are ignored, a successful response is still returned because the command didn't fail to write anything that it attempted to write.

Proprietary and Confidential ©2025 Vertiv Group Corp.

1.6 Authentication

The Avocent ACS800/8000 advanced console system RESTful API supports two different methods of authentication: JSON Web Token (JWT) and Basic Authentication.

1.6.1 JSON Web Token (JWT)

Using the JWT method, you can log in and authenticate using the /sessions/login resource, passing it a valid appliance username and password in the JSON body of the request. If successful, the appliance returns a JWT that must be included in the header of all subsequent requests as the "Authorization" key with a value of "Bearer <JWT>". This minimizes some of the appliance authentication overhead on each individual RESTful API call.

The JWT remains valid for 60 minutes. A GET request (or call) via the /sessions/refresh resource can be sent before the token expires to refresh and provide a new token.

NOTE: The username and password are transmitted as unencrypted plain text in the original /sessions/login request body, so it is recommended to use HTTPS for RESTful communications.

1.6.2 Basic authentication

Basic authentication takes a username/password pair and encodes it using base64. The resulting base64 value must then be included in every request header as the "Authorization" key with a value of "Basic <BASE64_VALUE>". After the request is authenticated by the appliance, the request is executed and the session terminated.

NOTE: The username and password are transmitted unencrypted (base64 is NOT a secure encryption) in every request, so it is recommended to use HTTPS for RESTful communications.

1.7 Document Conventions

1.7.1 Abbreviated URLs

To make this document more readable, the URLs in the examples are generally abbreviated to show only the portion of the URL after the /v1. The full URL is necessary when using the API. For example:

/system/info is shown instead of https://10.20.30.40:48048/api/v1/system/info

1.7.2 Examples

In the RESTful examples throughout this document, the request is shown in **bold** type, including both URL components and the message body. The response body is shown in normal type. For example:

```
POST /sessions/login {"username": "admin", "password": "avocent"}
        {
            "token":"eyJhbGciOiJIUzI1NiIsInR...fVzCM"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

This page intentionally left blank

Proprietary and Confidential ©2025 Vertiv Group Corp.

2 API Resources and Methods

The following table lists the resources the API provides.

Table 4.1 URL Descriptions

URL	Method	Description
SESSIONS		
/sessions/login	POST	Login with the username and password to create an API session and return a token in the JSON body.
/sessions/logout	POST	Exit and clean up the current session.
/sessions/refresh	GET	Using the token provided, obtain a new refreshed token which is returned in the JSON body response.
/sessions	GET	List of all active sessions.
/sessions/<ID>	GET DELETE	Returns details of specified session or kills the specified session.
SYSTEM		
/system/info	GET	Read basic system info such as serial number and type.
/system/reboot	POST	Reboot the appliance. Returns immediately.
/system/shutdown	POST	Shut down the appliance. Returns immediately.
/system/factoryDefault	POST	Reset the appliance to the factory defaults and reboots.
/system/firmware	GET	Returns the status of the system/firmware installation and download actions and whether they are in progress, failed, or successful, as well as firmware version information.
/system/firmware/version	GET	Returns version information of firmware, bootcode and date of build.
/system/firmware/download	POST	Download a firmware image file to the appliance using ftp, sftp or scp.
/system/firmware/install	POST	Install a previously downloaded firmware image file.
/system/firmware/downloaded	GET	Returns the version information of a previously downloaded firmware file.
/system/config	GET	Returns the status of the last system/config action and whether an action is in progress, failed, or successful.
/system/config/save	POST	Save the appliance configuration to a file.
/system/config/restore	POST	Restore the appliance configuration from a saved file.
/system/dateAndTime	GET PUT PATCH	Read and configure system time parameters.
/system/dateAndTime/testNtp	POST	Test and report on the NTP server settings.
/system/dateAndTime/timezones	GET	Returns a list of all recognized time zones.
/system/dateAndTime/timezone/custom	GET PUT PATCH	Read and configure a custom time zone.
/system/general	GET	Read and configure general system parameters, including online help,

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
	PUT PATCH	language, banner and so on.
/system/bootConfig	GET PUT PATCH	Read and configure the boot configuration parameters.
/system/certificate/download /system/certificate/generate /system/certificate/upload /system/certificate/apply	POST	Operations to generate, download, apply, or manage certificates.
/system/certificate	GET	Read certificate management status.
/system/integrity/generate /system/integrity/verify	POST	Generate and verify MD5 system configuration integrity.
/system/usage/memory /system/usage/flash	GET	Provides read-only system usage information.
SECURITY		
/security	GET PUT PATCH	Read and configure various security profile parameters.
/security/clearDSView	POST	Clear the certificate installed by the Vertiv™ management software.
NETWORK		
/network/settings	GET PUT PATCH	Read and configure various appliance specific network parameters.
/network/devices	GET	Returns information on all network devices.
/network/devices/<INTERFACE>	GET PUT PATCH	Read and configure network device-specific parameters: method (DHCP or static), IP address, netmask, gateway and so on.
/network/dhcpServerAddresses	GET POST	Read and add IP address assignments made by the DHCP server.
/network/dhcpServerAddresses/<IPADDRESS>	GET PUT PATCH DELETE	Read and configure the individual IP address assignments made by the DHCP server.
/network/dhcpServerSettings	GET PUT PATCH	Read and configure the DHCP server settings.
/network/staticRoutes/ipv4 /network/staticRoutes/ipv6	GET POST	Read and create static route table entries.

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
/network/staticRoutes/ipv4/<ID> /network/staticRoutes/ipv6/<ID>	GET PUT PATCH DELETE	Read, edit, or delete an individual static route table entry.
/network/hosts	GET POST	Read and create network hosts entries.
/network/hosts/<IPADDRESS>	GET PUT PATCH DELETE	Read and configure an individual network host entry.
/network/firewall/ipv4 /network/firewall/ipv6	GET POST	Read or create chains in an IP filter table.
/network/firewall/ipv4/<CHAIN> /network/firewall/ipv6/<CHAIN>	GET PUT PATCH DELETE	Read or configure a specific chain in an IP filter table.
/network/firewall/ipv4/<CHAIN>/rules /network/firewall/ipv6/<CHAIN>/rules	GET POST	Read or create rules in an IP filter table chain.
/network/firewall/ipv4/<CHAIN>/rules/<INDEX> /network/firewall/ipv6/<CHAIN>/rules/<INDEX>	GET PUT PATCH DELETE	Read or configure a specific rule in an IP filter table chain.
/network/firewall/ipv4/<NAME>/rules/<INDEX>/move /network/firewall/ipv6/<NAME>/rules/<INDEX>/move	POST	Move a firewall rule up or down the list of rules.
/network/certificates/settings	GET PUT PATCH	Read or configure the network certificate management settings.
/network/certificates/status	GET	Read the status of the network certificate management process.
/network/certificates/root	GET	Read a list of the existing root certificate file names.
/network/certificates/root/<NAME>	GET DELETE	Read the details of or delete a specific root certificate.
/network/certificates/root/download	POST	Download a root certificate from a remote server.
/network/certificates/root/upload	POST	Upload a root certificate directly via the RestAPI.
/network/ipsec/certificates	GET	Read a list of the existing IPSec certificate file names.
/network/ipsec/certificates/<NAME>	DELETE	Delete a specific IPSec certificate file.
/network/ipsec/certificates/download	POST	Download an IPSec certificate from a remote server.
/network/ipsec/connections	GET POST	Read all connections or create a new IPSec connection.
/network/ipsec/connections/<NAME>	GET PUT	Read or configure a specific IPSec connection.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
	PATCH DELETE	
/network/ipsec/connections/<NAME>/connect /network/ipsec/connections/<NAME>/disconnect /network/ipsec/connections/<NAME>/ping	POST	Perform actions on a specific IPSec connection.
/network/ipsec/connections/<NAME>/diagnostics	GET	Read diagnostic information for a specific IPSec connection.
/network/gre/tunnels	GET POST	Read all GRE tunnels or create a new GRE tunnel.
/network/gre/tunnels/<NAME>	GET PUT PATCH DELETE	Read or configure a specific GRE tunnel.
/network/gre/tunnels/<NAME>/connect /network/gre/tunnels/<NAME>/disconnect /network/gre/tunnels/<NAME>/ping	POST	Perform actions on a specific GRE tunnel.
/network/gre/tunnels/<NAME>/diagnostics	GET	Read diagnostic information for a specific GRE tunnel.
/network/snmp	GET POST	Read all SNMP entries or create a new SNMP entry.
/network/snmp/<ID>	GET PUT PATCH DELETE	Read or configure a specific SNMP entry.
/network/snmp/system	GET PUT PATCH	Read or configure SNMP system settings.
SERIAL PORTS		
/serialPorts	GET	Read various serial port parameters: status, pinout, parity, profile, and so on.
/serialPorts/<PORT>	GET PUT PATCH	Read and configure various serial port parameters for individual serial ports.
/serialPorts/<PORT>/enable /serialPorts/<PORT>/disable /serialPorts/<PORT>/clone /serialPorts/<PORT>/discoverNow /serialPorts/<PORT>/resetToFactory	POST	Perform actions on a specific serial port.
/serialPorts/<PORT>/alerts	GET POST	Read all alerts or create a new alert for a specific serial port.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
/serialPorts/<PORT>/alerts/clear /serialPorts/<PORT>/alerts/deleteAny	POST	Perform actions on all alerts for a specific serial port.
/serialPorts/<PORT>/alerts/<ID>	DELETE	Delete an individual alert for a specific serial port.
/serialPorts/<PORT>/power	GET POST	Read or create power outlets or groups for a specific serial port.
/serialPorts/<PORT>/power/<ID>	DELETE	Delete a specific power outlet or group for a specific serial port.
AUXILIARY PORTS AND MODEM		
/auxPorts/	GET	Read auxiliary port information. DEPRECATED, use /modems.
/auxPorts/<NAME>	GET PUT PATCH	Read or configure specific auxiliary port information. DEPRECATED, use /modems.
/modems	GET	Read information for all modems.
/modems/<NAME>	GET PUT PATCH	Read or configure a specific modem.
/modems/<NAME>/signalCheck /modems/<NAME>/registration	POST	Perform actions on a specific modem.
CAS PROFILE		
/casProfile	GET PUT PATCH	Read and configure the CAS profile parameters.
/casProfile/probeStrings	GET POST	Read all or create a new CAS profile probe string.
/casProfile/probeStrings/<INDEX>	GET DELETE	Read or delete a specific probe string.
/casProfile/matchStrings	GET POST	Read all or create a new CAS profile match string.
/casProfile/matchStrings/<INDEX>	GET DELETE	Read or delete a specific match string.
/casProfile/commands	GET POST	Read all or create a new auto discovery command.
/casProfile/commands/<NAME>	GET PUT PATCH DELETE	Read, edit, or delete a specific auto discovery command.
/casProfile/autoAnswer	GET POST	Read all or create a new auto answer string pair.
/casProfile/autoAnswer/<INDEX>	GET DELETE	Read or delete a specific auto answer string pair.
/casProfile/poolOfPorts	GET POST	Read all or create a new pool of ports.

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
/casProfile/poolOfPorts/<NAME>	GET PUT PATCH DELETE	Read, edit, or delete a specific pool of ports.
DIAL-IN PROFILE		
/dialinProfile	GET PUT PATCH	Read or configure the Dial-In Profile settings.
/dialinProfile/callbackUsers	GET POST	Read all or add a new callback user.
/dialinProfile/callbackUsers/<NAME>	GET PUT PATCH DELETE	Read, configure, or delete a specific callback user.
/dialinProfile/callerId	GET POST	Read all or add a new caller ID number.
/dialinProfile/callerId/<INDEX>	GET DELETE	Read or delete a specific caller ID number.
/dialinProfile/chapSecrets	GET POST	Read all or add a new chap secret.
/dialinProfile/chapSecrets/<NAME>	GET PUT PATCH DELETE	Read, configure, or delete a specific chap secret.
/dialinProfile/pppOtpUsers	GET POST	Read all or add a new PPP OTP user.
/dialinProfile/pppOtpUsers/<NAME>	GET DELETE	Read, configure, or delete a specific PPP OTP user.
PLUGGABLE DEVICES		
/pluggableDevices /pluggableDevices/<NAME>	GET	Returns a list of attached pluggable devices and their details or returns details on the named pluggable device.
/pluggableDevices/<NAME>/eject	POST	Eject the specified device so that it is safe to remove.
/pluggableDevices/<NAME>/delete	POST	Delete the specified device after unplugging it.
/pluggableDevices/<NAME>/setConsole	POST	Set up the specified pluggable device as a console port.
AUTHENTICATION		
/authentication	GET PUT PATCH	Read and configure general appliance authentication parameters.
/authentication/dsview /authentication/duo	GET PUT PATCH	Read and configure authentication parameters specific to each type of authentication server.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
/authentication/kerberos /authentication/ldap /authentication/radius /authentication/tacacs		
USERS AND GROUPS		
/users	GET POST	Read all or add a new local user to the appliance.
/users/<NAME>	GET PUT PATCH POST DELETE	Read, configure, or delete a specific local user account.
/users/<NAME>/unlock	POST	Unlock the user account if it is locked.
/users/<NAME>/disable	POST	Disable the user account.
/users/<NAME>/enable	POST	Enable the user account.
/users/passwordRules	GET PUT PATCH	Read or configure the appliance's password rules.
/groups	GET POST	Read all or add a new user group.
/groups/<NAME>	GET PUT PATCH DELETE	Provides the ability to view and edit group settings, as well as add new users and delete existing users from groups.
ACCESS RIGHTS		
/accessRights/users/<NAME>/serialPorts /accessRights/groups/<NAME>/serialPorts	GET POST	Configure the individual user's or group's access rights to serial ports.
/accessRights/users/<NAME>/serialPorts/<PORT> /accessRights/groups/<NAME>/serialPorts/<PORT>	GET PUT PATCH DELETE	Configure the individual user's or group's access rights to serial ports.
/accessRights/users/<NAME>/appliance /accessRights/groups/<NAME>/appliance	GET PUT PATCH	Configure the individual user's or group's access rights to the appliance.
/accessRights/users/<NAME>/pdus /accessRights/users/<NAME>/ups /accessRights/groups/<NAME>/pdus /accessRights/groups/<NAME>/ups	GET POST	Configure the individual user's or group's access rights to PDU or UPS devices.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
/accessRights/users/<NAME>/pdus/<ID> /accessRights/users/<NAME>/ups/<ID> /accessRights/groups/<NAME>/pdus/<ID> /accessRights/groups/<NAME>/ups/<ID>	DELETE	Delete the individual user's or group's access rights for a specific PDU or UPS devices.
/accessRights/users/<NAME>/outlets /accessRights/users/<NAME>/upsOutletGroups /accessRights/groups/<NAME>/outlets /accessRights/groups/<NAME>/upsOutletGroups	GET POST DELETE	Configure the individual user's or group's access rights to individual PDU or UPS outlets or outlet groups.
/accessRights/users/<NAME>/outlets/<ID> /accessRights/users/<NAME>/upsOutletGroups/<ID> /accessRights/groups/<NAME>/outlets/<ID> /accessRights/groups/<NAME>/upsOutletGroups/<ID>	DELETE	Delete the individual user's or group's access rights for specific outlets or outlet groups.
/accessRights/dsview	GET PUT PATCH	Provides the ability to configure the access rights for the Vertiv™ management software.
EVENTS AND LOGS		
/events	GET	Read the settings for all appliance events.
/events/<ID>	GET PUT PATCH	Read or configure the settings for a specific appliance event.
/events/applianceLogging	GET PUT PATCH	Read or configure the appliance logging parameters.
/events/dataBuffering	GET PUT PATCH	Read or configure the data buffering settings.
/events/dsview	GET PUT PATCH	Read or configure the event parameters for the Vertiv™ management software.
/events/snmp	GET PUT PATCH	Read or configure the event SNMP parameters.
/events/sms	GET PUT PATCH	Read or configure the event SMS parameters.
/events/email	GET PUT PATCH	Read or configure the event destination email parameters.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
/events/syslog	GET PUT PATCH	Read or configure the event syslog parameters.
/events/trapForward	GET POST	Read all or add a new Trap Forward entry.
/events/trapForward/<INDEX>	GET PUT PATCH DELETE	Read, configure or delete a specific Trap Forward entry.
/events/trapForward/settings	GET PUT PATCH	Read or configure trap forward settings.
/events/test	POST	Generate a test event to test the event destinations.
POWER MANAGEMENT		
/power/login	GET PUT PATCH	Read or configure passwords for the various brands of power devices.
/power/networkPdus	GET POST	Read all or add a new network PDU.
/power/networkPdus/<IPADDRESS>	GET PUT PATCH DELETE	Read, configure, or delete a specific network PDU.
/power/networkUps	GET POST	Read all or add a new network UPS.
/power/networkUps/<IPADDRESS>	GET PUT PATCH DELETE	Read, configure, or delete a specific network UPS.
/power/pdus	GET	Read information about all PDU devices.
/power/pdus/<NAME>	GET PUT PATCH	Read or configure settings for a specific PDU.
/power/pdus/<NAME>/banks	GET	Read information about all the banks on a specific PDU.
/power/pdus/<NAME>/banks/<INDEX>	GET PUT PATCH	Read or configure a specific bank on an individual PDU.
/power/pdus/<NAME>/banks/<INDEX>/resetValues	POST	Reset the electrical monitoring values on a specific bank on an individual PDU.
/power/pdus/<NAME>/cycle /power/pdus/<NAME>/factoryDefaults	POST	Perform a specified action on an individual PDU.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
/power/pdus/<NAME>/off /power/pdus/<NAME>/on /power/pdus/<NAME>/firmwareDownload /power/pdus/<NAME>/firmwareInstall /power/pdus/<NAME>/reboot /power/pdus/<NAME>/refresh /power/pdus/<NAME>/rename /power/pdus/<NAME>/resetValues		
/power/pdus/<NAME>/inlets	GET	Read information about all the inlets on a specific PDU.
/power/pdus/<NAME>/inlets/<INDEX>	GET PUT PATCH	Read or configure a specific inlet on an individual PDU.
/power/pdus/<NAME>/outlets	GET	Read information about all the outlets on a specific PDU.
/power/pdus/<NAME>/outlets/<INDEX>	GET PUT PATCH	Read or configure a specific outlet on an individual PDU.
/power/pdus/<NAME>/outlets/<INDEX>/cycle /power/pdus/<NAME>/outlets/<INDEX>/lock /power/pdus/<NAME>/outlets/<INDEX>/off /power/pdus/<NAME>/outlets/<INDEX>/on /power/pdus/<NAME>/outlets/<INDEX>/resetValues /power/pdus/<NAME>/outlets/<INDEX>/unlock	POST	Perform a specified action on a specific outlet of an individual PDU.
/power/pdus/<NAME>/phases	GET	Read all phases on the specific PDU.
/power/pdus/<NAME>/phases/<INDEX>	GET PUT PATCH	Read and configure the specified phase on a specific PDU.
/power/pdus/<NAME>/phases/<INDEX>/resetValues	POST	Reset the values of the specified phase on a specific PDU.
/power/pdus/<NAME>/sensors	GET	Read all sensor information on the specified PDU.
/power/pdus/<NAME>/sensors/<INDEX>	GET PUT PATCH	Read or configure the specific sensor information the specified PDU.
/power/pdus/<NAME>/sensors/<INDEX>/resetValues	POST	Reset the values of the specified sensor on a specific PDU.
/power/ups	GET	Read information about all UPS devices.
/power/ups/<NAME>	GET	Read and configure the specific UPS device.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
	PUT PATCH	
/power/ups/<NAME>/outletGroups	GET	Read information about all the outlet groups on a specific UPS.
/power/ups/<NAME>/outletGroups/<INDEX>	GET	Read information about an individual outlet group on a specific UPS.
/power/ups/<NAME>/outletGroups/<INDEX>/cycle /power/ups/<NAME>/outletGroups/<INDEX>/off /power/ups/<NAME>/outletGroups/<INDEX>/on	POST	Perform a specified action on a specific outlet group of a specified UPS.
/power/ups/<NAME>/outputCycle /power/ups/<NAME>/outputOff /power/ups/<NAME>/outputOn /power/ups/<NAME>/rename /power/ups/<NAME>/resetPowerStats /power/ups/<NAME>/restoreName /power/ups/<NAME>/silenceAlarm /power/ups/<NAME>/testBattery	POST	Perform a specified action on an individual UPS.
/power/outletGroups	GET POST	Read all or add a new outlet group.
/power/outletGroups/<NAME>	GET DELETE	Read or delete an individual outlet group.
/power/outletGroups/<NAME>/cycle /power/outletGroups/<NAME>/off /power/outletGroups/<NAME>/on	POST	Perform a specified action on a specific outlet group.
/power/outletGroups/<NAME>/outlets	GET POST	Read all outlets or add a new outlet to a specific outlet group.
/power/outletGroups/<NAME>/outlets/<ID>	DELETE	Delete an individual outlet from an outlet group.
SENSORS		
/sensors/internal	GET PUT PATCH	Read or configure the internal sensors.
/sensors/1Wire	GET	Read all of the 1-Wire sensors.
/sensors/1Wire/<HEXADDR>	GET DELETE PATCH PUT	Read and configure individual 1-Wire sensors.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
/sensors/1Wire/refresh	POST	Refresh the list of 1-Wire sensors.
/sensors/digitalIn	GET	Read the digital in sensor data.
/sensors/digitalIn/<ID>	GET PUT PATCH	Read and configure individual digital in sensors.
/sensors/pdu	GET	Read all of the PDU sensor data.
/sensors/pdu/<NAME>/reset	POST	Reset the PDU sensor data of a specific PDU.
DIGITAL OUT		
/digitalOut	GET	Read all the digital out port information.
/digitalOut/<ID>	GET PUT PATCH	Read and configure individual digital out ports.
MONITORING		
/monitoring/network/devices	GET	Read the status of all the network devices.
/monitoring/network/devices/<INT>	GET	Read the status of a specific network device.
/monitoring/network/routingTables/ipv4 /monitoring/network/routingTables/ipv6	GET	Read the ipv4 or ipv6 routing tables.
/monitoring/serialPorts	GET	Read the state of the serial ports including transmission counts.
/monitoring/serialPorts/<ID>	GET DELETE	Read the state of an individual serial port or clear the counters to zero.
/monitoring/scheduledTasks	GET POST	Read all scheduled tasks or create a new scheduled task to be run on the appliance.
/monitoring/scheduledTasks/<NAME>	GET PUT PATCH DELETE	Read, configure, or delete a specific scheduled task.
/monitoring/scheduledTasks/<NAME>/runNow	POST	Run the specified scheduled task.
/monitoring/autoDiscovery	GET	Read the auto discovery process output of all configured serial ports.
/monitoring/autoDiscovery/<PORT>	GET	Read the auto discovery process output of a specific serial port.
/monitoring/autoDiscovery/<PORT>/discoverNow	POST	Run auto discovery on the specified port immediately.
/monitoring/fipsMode	GET	Monitor services which are in FIPS mode.
/monitoring/callerIdLog	GET	Read the Caller ID log.
/monitoring/callerIdLog/clearLog	POST	Clear the Caller ID log.
/monitoring/modemPppdLog	GET	Read the Modem PPPD log.
/monitoring/modemPppdLog/clearLog	POST	Clear the Modem PPPD log.
/monitoring/ipsec	GET	Read the status of all IPSec connections.

Table 4.1 URL Descriptions (continued)

URL	Method	Description
/monitoring/ipsec/<NAME>	GET	Read the status of a specific IPSec connection.
/monitoring/zeroTouchLog	GET	Read the Zero Touch provisioning log.
/monitoring/zeroTouchLog/clearLog	POST	Clear the Zero Touch log.
ACCESS		
/access/serialPorts	GET	Returns a list of ports and devices that can be accessed, along with their status.
/access/serialPorts/<PORT>	GET	Returns the status information of a specific serial port that can be accessed.
/access/serialPorts/<PORT>/on /access/serialPorts/<PORT>/off /access/serialPorts/<PORT>/cycle	POST	Provides actions to control power of the selected device if configured.
MISCELLANEOUS		
/resources	GET	Returns a list of available API resources.
/changePassword	POST	Set the initial password or change the password of the local, authenticated RestAPI user.

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.1 Sessions

2.1.1 /sessions/login

Use this resource to establish a connection using the username and password provided in the JSON body with the configured authentication of the appliance. A web token is returned to be sent in the header of ALL subsequent requests as the "Authorization" key with a value of "Bearer <TOKEN>".

An alternative to this log-in session is to send a base64 encoded username/password pair in every API request, using an "Authorization" key with a value of "Basic <BASE64_VALUE>". For more information, refer to Authentication on page 191.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
username	Valid username of an account on the appliance (root, admin and so on).
password	Valid password for the specified username.

Response Body

Parameter	Description
token	Valid authorization token to use on subsequent requests.

Response Codes

200	OK
400	Bad Request
401	Not Authorized

Examples

```
POST /sessions/login {"username": "admin", "password": "avocent"}
        {
            "token":"eyJhbGciOiJIUzI1NiIsInR...fVzCM"
        }
```

```
POST /sessions/login {"username": "bad", "password": "bad"}
        {
            "error": {
                "code": "AE017",
                "message": "user authentication failed"
            }
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.1.2 /sessions/logout

Use this resource to disconnect the current connection, free any web token or other session authentication, and clean up the session.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
logout	Status of request, typically OK.
username	The username that was logged out.

Response Codes

200	OK
400	Bad Request
401	Not Authorized

Examples

```
POST /sessions/logout
        {
            "logout": "OK",
            "username": "root"
        }
```

2.1.3 /sessions/refresh

Use this resource to refresh the current connection token to keep it active.

Methods

GET

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
token	Valid authorization token to use on subsequent requests.

Response Codes

200	OK
400	Bad Request
401	Not Authorized

Examples

```
GET /sessions/refresh
        {
            "token":"eyJhbGciOiJIUzI1NiIsInR...fVzCM"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.1.4 /sessions[/<ID>]

Use this resource to view all active sessions and session information and to kill specific sessions.

Methods

GET, DELETE

Parameters

Fields query is supported for all body parameters.

Filtering query is supported for all parameters, except for creationTime on GET /sessions.

Parameter	Description
ID	Unique session ID number (integer).

Request Body

None.

Response Body

Parameter	Description
id	Unique session ID number (integer).
user	Username which owns the session.
clientIp	IP address of the client which originated the session.
creationTime	Date and time on which the session was created.
sessionType	Type of session: ssh / telnet / console / http / https / raw / unknown
connectionType	Type of connection: serial / wmi / cli / api / unknown
targetName	Name of the target of the session, or blank if the appliance.
parentId	Unique session ID number.

Response Codes

200	OK
400	Bad Request
404	Not Found

Examples

GET /sessions

```
{
    "sessions": [
        {
            "id": 24,
            "user": "root",
            "clientIp": "10.20.30.40",
            "creationTime": "Tue 01 Aug 2017 07:15:36 PM UTC",
            "sessionType": "http",
            "connectionType": "wmi",
            "target": "",
            "parentId": "",
        },
        {
            "id": 43,
            …
        },
        …
    ]
}
```

GET /sessions/100

```
{
    "id": 100,
    "user": "root",
    "clientIp": "10.20.30.40",
    "creationTime": "Tue 01 Aug 2017 07:15:36 PM UTC",
    "sessionType": "http",
    "connectionType": "wmi",
    "target": "",
    "parentId": "",
}
```

DELETE /sessions/101

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2 System

2.2.1 /system/info

Use this resource to access read-only system information about the appliance's identity, versions, power and CPU information.

Methods

GET

Parameters

Fields query is supported for all body parameters.

Request Body

None.

Response Body

Parameter	Description
serialNumber	Serial number assigned to the appliance at the factory.
type	Description of the type of unit, including the model number with port count, power supplies and modem presence. Example: ACS8048 with single power supply
bootcode	Version number of the installed bootcode. Example: 1.18
firmware	Full version number of the installed firmware. Example: 1.3.75.2779+551+28+11
firmwareDate	Date of the installed firmware. Example: Sep 1 2016 – 04:07:14
bootedFrom	Identifies whether the appliance is currently booted from hardware (internal Flash) or network.
powerSupply1	Status of power supply 1: on/off
powerSupply2	Status of power supply 2, if present: on/off
cpu	Description of the CPU. Example: ARMv7 Processor rev 0 (v71)
cores	Number of cores in the CPU (integer): 2

Response Codes

200	OK
400	Bad request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /system/info


```
{
    "serialNumber": "1234567890",
    "type": "ACS8048 with single power supply",
    "bootcode": "1.18",
    "firmware": "1.3.75.2779+551+28+11",
    "firmwareDate": "Sep 1 2016 - 04:07:14",
    "bootedFrom": "hardware",
    "powerSupply1": "on",
    "cpu": "ARMv7 Processor rev 0 (v7l)",
    "cores": 2
}
```


 Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.2 /system/reboot

Use this resource to reboot the appliance.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “initiated reboot” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /system/reboot
        {
            "status": "initiated reboot"
        }
```

2.2.3 /system/shutdown

Use this resource to shut down the appliance.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “initiated shutdown” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /system/shutdown
        {
            "status": "initiated shutdown"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.4 /system/factoryDefault

Use this resource to restore the appliance to the factory default settings and reboot the appliance.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “initiated factoryDefault” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /system/factoryDefault
        {
            "status": "initiated factoryDefault"
        }
```

2.2.5 /system/firmware

Use this resource to view the status of the firmware's download and installation processes, including the firmware image information.

Methods

GET

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Firmware status: none / download in progress / download successful / download failed / install in progress / install successful / install failed
version	Full version number of the firmware file. Example: 1.2.9.2449+540+23+11
date	Date of the build of the firmware file. Example: 03/01/17

Response Codes

200	OK
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /system/firmware

```
{
    "status": "download in progress"
}
```

GET /system/firmware

```
{
    "status": "download successful",
    "version": "1.2.9.2449+540+23+11",
    "date": "03/01/17"
}
```

GET /system/firmware

```
{
    "error": {
        "code": "AE011",
        "message": "upgrade failed",
        "detail": "Failed to open firmware image."
    }
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.6 /system/firmware/version

Use this resource to access information about the currently installed and running firmware, including build date and version numbers of various components.

Methods

GET

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
version	Full version number of the installed firmware. Example: 1.3.75.2779+551+28+11
bootVersion	Version number of the installed bootcode. Example: 1.18
date	Date of the firmware build. Example: Aug 12 2017 – 09:12:24

Response Codes

200	OK
400	Bad Request

Examples

```
GET /system/firmware/version
        {
            "version": "1.3.75.2779+551+28+11",
            "bootVersion": "1.18",
            "date": "Aug 12 2017 - 09:12:24"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.7 /system/firmware/download

Use this resource to download the specified firmware file in preparation for updating the firmware. By default, the action does not return until the file download has either completed or failed. Depending on the network speed, this could take a couple minutes.

If nonblocking is included in the flags parameter, the download begins and the action returns immediately. A GET request to the /system/firmware resource must then be used to monitor for command completion.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
protocol	Specify the protocol to use to download the firmware: ftp / scp / sftp / https
ipAddress	IP address of the remote server from which to download the file.
username	Username to access the remote server.
password	Password to access the remote server.
directory	Directory path on the remote server, typically relative to the ftp root directory.
filename	Filename of the firmware file on the remote server.
flags	Supported flags: nonblocking Default: blank

Response Body

Parameter	Description
status	Status result, typically “download initiated” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /system/firmware/download
        {
                "protocol": "ftp",
                "ipAddress": "10.20.30.80",
                "username": "anonymous",
                "password": "anonymous",
                "directory": "pub/firmware/",
                "filename": "firmware_acs8_1_2_9.fl"
        }
```

Response is:

```
        {
            "status": "download successful",
            "firmware": {
                "version": "1.2.9.2449+540+23+11",
                "date": "03/01/17"
            }
        }
```

```
POST /system/firmware/download
        {
                "protocol": "ftp",
                "ipAddress": "10.20.30.80",
                "username": "anonymous",
                "password": "anonymous",
                "directory": "pub/firmware/",
                "filename": "firmware_acs8_1_2_9.fl",
                "flags": "nonblocking"
        }
```

Response is:

```
        {
            "status": "download initiated",
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.8 /system/firmware/install

Use this resource to install a previously downloaded firmware image into flash memory. This request does not return until the installation is complete, which may take up to two minutes.

If nonblocking is included in the flags parameter, the installation begins and the action returns immediately. A GET request to the /system/firmware resource must then be used to monitor for command completion.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
flags	Optional supported flags: **nonblocking** Default: blank

Response Body

Parameter	Description
status	Status result, typically “install successful” on success.
firmware:	
version	Full version number of the installed firmware. Example: 1.2.9.2449+540+23+11
date	Date of the firmware build. Example: 03/01/17

Response Codes

200	OK
400	Bad Request

Examples

```
POST /system/firmware/install
{
    "status": "install successful",
    "firmware": {
        "version": "1.2.9.2449+540+23+11",
        "date": "03/01/17"
    }
}
```

```
POST /system/firmware/install
{
    "flags": "nonblocking"
}
```

Response is:

```
{
    "status": "install initiated"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.9 /system/firmware/downloaded

Use this resource to access information about a firmware image that has previously been downloaded to the appliance.

NOTE: This resource has been deprecated. The same functionality is now available via the /system/firmware resource.

Methods

GET

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
version	Full version number of the firmware file. Example: 1.2.9.2449+540+23+11
date	Date of the build of the firmware file. Example: 03/01/17

Response Codes

200	OK
400	Bad Request

Examples

```
GET /system/firmware/downloaded
        {
            "version": "1.2.9.2449+540+23+11",
            "date": "03/01/17"
        }
```

2.2.10 /system/config

Use this resource to view the status of the last configuration action.

Methods

GET

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Configuration status: none / save in progress / save successful / save failed / restore in progress / restore successful / restore failed

Response Codes

200	OK
400	Bad Request

Examples

```
GET /system/config
        {
            "status": "save in progress"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.11 /system/config/save

Use this resource to save the system configuration of the appliance. If used without the nonblocking flag, this request does not return until the save is complete. It may take several minutes, depending on the format.

If nonblocking is included in the flags parameter, the configuration save begins and the action returns immediately. A GET request to the /system/config resource must then be used to monitor for command completion.

NOTE: The API does not support XML format.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
format	Format used to save the configuration: **cli** / **compressed** Default: cli
where	Where to save the file, either to the local appliance or a remote server: **local** / **remote**
protocol	Protocol used to transfer the file to a remote server: **ftp** / **scp** / **sftp**
ipAddress	IP address of the remote server.
username	Username of the account used to log in to the remote server.
password	Password for the specified username to log into the remote server. Default: "anonymous"
directory	Directory where the file is to be written. If the directory starts with "/", it is considered an absolute path. Otherwise, the directory is relative to /mnt/hdUser/backup for local, which is the default location of local configuration files, or relative to the specified protocol's configured base directory on the remote server.
filename	Filename to use for the saved configuration file.
scrub	Controls whether to scrub potentially sensitive data from the saved configuration file: **enabled** / **disabled**. **NOTE: This option should only be enabled when saving a configuration being distributed to a third party such as Vertiv Technical Support. This is for debugging purposes only, and the saved configuration will not have some of the key configuration elements necessary for a full restoration.**
flags	Supported flags: nonblocking Default: blank

Response Body

Parameter	Description
status	Status result string indicating whether the operation was successful and where the configuration file was saved.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /system/config/save
        {
            "format": "cli",
            "where": "remote",
            "protocol": "ftp",
            "ipAddress": "10.20.30.70",
            "username": "anonymous",
            "password": "anonymous",
            "directory": "pub",
            "filename": "myconfig.cli",
            "scrub": "disabled",
            "flags":""
        }
```

Response is:

```
        {
            "status": "backup configuration saved as cli config filename: pub/myconfig.cli"
        }
```

```
POST /system/config/save
        {
            "format": "cli",
            "where": "remote",
            "protocol": "ftp",
            "ipAddress": "10.20.30.70",
            "username": "anonymous",
            "password": "anonymous",
            "directory": "pub",
            "filename": "myconfig.cli",
            "scrub": "disabled",
            "flags":"nonblocking"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

Response is:

```
{
        "status": "save configuration initiated"
}
```

2.2.12 /system/config/restore

Use this resource to restore the appliance configuration from a specified file. If used without the nonblocking flag, this request does not return until the restore is complete. This may take several minutes, depending on the format.

If nonblocking is included in the flags parameter, the configuration restore begins and the action returns immediately. A GET request to the /system/config resource must then be used to monitor for command completion.

NOTE: The API does not support XML format.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
where	Where the configuration file is located: **local** / **remote**
protocol	Protocol used to retrieve the remote file: **ftp** / **scp** / **sftp**
ipAddress	IP address of the remote server.
username	Username of the account to use on the remote server.
password	Password for the specified username on the remote server. Default: "anonymous"
directory	Directory where the file is located. If the directory starts with "/", it is considered an absolute path. Otherwise, the directory is relative to /mnt/hdUser/backup for local, which is the default location of the local configuration files, or relative to the ftp/scp/sftp configured base directory on the remote server.
filename	Name of the configuration file to restore.
flags	Supported flags: nonblocking Default: blank

Response Body

Parameter	Description
status	Status result string indicating whether the operation was successful and from where the configuration was restored.

Response Codes

200	OK
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
POST /system/config/restore
        {
            "where": "remote",
            "protocol": "ftp",
            "ipAddress": "10.20.30.70",
            "username": "anonymous",
            "password": "anonymous",
            "directory": "pub",
            "filename": "myconfig.cli",
            "flags": ""
        }
```

Response is:

```
        {
            "status": "backup configuration restored from cli config filename:
            pub/myconfig.cli"
        }
```

The following resource restores from the local file /mnt/hdUser/backup/myconfig.cli, which is sitting in the default local directory for backup configuration files. The directory parameter is not actually needed in this case, as it is assumed to be blank if not provided.

```
POST /system/config/restore
    {
        "where": "local",
        "directory": "",
        "filename": "myconfig.cli"
    }
```

Response is:

```
    {
        "status": "backup configuration restored from cli config filename: myconfig.cli"
    }
```

2.2.13 /system/integrity/generate

Use this resource to generate an MD5 tag for the running configuration; this request takes no parameters. It returns an MD5 tag string in the following form: bf86785ade6fa3d5999f2f85752a4fd

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
md5Tag	The unique MD5 tag integrity value generated for the current configuration.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /system/integrity/generate
```

Response is:

```
{
    "md5Tag": "bf86785ade6fa3d5999f2f85752a4fd"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.14 /system/integrity/verify

Use this resource to verify the running configuration's MD5 against a supplied MD5 value. If no MD5 is supplied, the current running configuration is compared against the last generated MD5.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
md5Tag	MD5 tag to compare against the MD5 of the running system. Example: "bf86785ade6fa3d5999f2f85752a4fd"

Response Body

Parameter	Description
integrityStatus	Status string containing the result of the configuration integrity check: Unchanged / Modified

Response Codes

200	OK
400	Bad Request

Examples

```
POST /system/integrity/verify '{"md5Tag":"badbadbadbadbadbadbadbadbadd"}'
```

Response is:

```
{
    "integrityStatus": "Modified"
}
```

2.2.15 /system/certificate/generate

Use this resource to generate a self-signed certificate or a Certificate Signing Request (CSR) for the appliance from the information provided. The resource does not return until the certificate generation has completed or failed. If nonblocking is included in the flags parameter, the generation begins and the action returns immediately. The /system/certificate resource must then be used to monitor for command completion.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
type	Type of request: **cert** / **csr** Default: “cert”
country	Country Name (two letter code). Example: "us"
state	State or Province name (full name). Example: "washington"
city	City or locality name. Example: "seattle"
organization	Organization or company name.
unit	Organizational unit or section name.
commonName	Server FQDN or another name.
subjectAltNames	Subject Alternative Names in a comma separate list including their type. Names without a type are treated as DNS: type. The following character set is permitted: "a-zA-z0-9.-:*@/+". Example: DNS:vertiv.com,IP:1.2.3.4.
emailAddress	Email address.
netscapeComment	Comment to display in Netscape's comment listbox.
newKey	Used only when type is “csr” to indicate whether a new key should be generated first. **enabled** / **disabled**
passphrase	Passphrase to protect private key file.
flags	Supported flags: **nonblocking** Default: blank

Response Body

Parameter	Description
status	Status result, typically “certificate generation successful” on success.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /system/certificate/generate
        {
            "country": "us",
            "state": "washington",
            "city": "seattle",
            "organization": "",
            "unit": "",
            "commonName": "mycertificate",
            "emailAddress": "me@myemail.com",
            "netscapeComment": "",
            "passphrase": ""
        }
```

Response is:

```
        {
            "status": "certificate generation successful",
        }
```

```
POST /system/certificate/generate
        {
            "type": "csr",
            "country": "us",
            "state": "washington",
            "city": "seattle",
            "organization": "",
            "unit": "",
            "commonName": "mycertificate",
            "subjectAltNames": "DNS:vertiv.com,IP:1.2.3.4",
            "emailAddress": "me@myemail.com",
            "netscapeComment": "",
            "newKey": "enabled",
            "passphrase": "",
            "flags": "nonblocking"
        }
```

Response is:

```
        {
            "status": "certificate generation initiated",
        }
```

2.2.16 /system/certificate/download

Use this resource to download a certificate from the specified server information. The resource does not return until the file download is complete or fails. Depending on the network speed, this could take a couple minutes.

If nonblocking is included in the flags parameter, the download begins and the action returns immediately. The /system/certificate resource must then be read to monitor for command completion.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
protocol	Specifies the protocol to use to download file: **ftp** / **scp** / **sftp**
ipAddress	The IP address of the remote server from which to download the file.
username	Username to access the remote server.
password	Password to access the remote server.
directory	Directory path on the remote server, typically relative to the ftp root directory.
filename	Filename of the certificate file on the remote server.
passphrase	Passphrase if required for an embedded key file.
flags	Supported flags: **nonblocking**

Response Body

Parameter	Description
status	Status result, typically “certificate download successful” on success.

Response Codes

200	OK
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
POST /system/certificate/download
        {
            "protocol": "ftp",
            "ipAddress": "10.20.30.80",
            "username": "anonymous",
            "password": "anonymous",
            "directory": "pub/certificate/",
            "filename": "mycertificate"
        }
```

Response is:

```
        {
            "status": "certificate download successful",
        }
```

```
POST /system/certificate/download
        {
            "protocol": "ftp",
            "ipAddress": "10.20.30.80",
            "username": "anonymous",
            "password": "anonymous",
            "directory": "pub/certificate/",
            "filename": "mycertificate"
            "flags": "nonblocking"
        }
```

Response is:

```
        {
            "status": "certificate download initiated",
        }
```

2.2.17 /system/certificate

Use this resource to report the status of a previous certificate action and to read and return the last generated or downloaded certificate or generated Certificate Signing Request (CSR).

Methods

GET

Parameters

Fields query is supported for all body parameters.

Format query is supported and applies to the data parameter. Formats include text and base64.

Request Body

None.

Response Body

Parameter	Description
status	Certificate activity status: none / in progress / successful / failed
type	Type of certificate information: cert / csr / download / upload
data	Data for the certificate or CSR. CSR is returned in raw data format for writing to a .pem file. Certificate information is returned as a text summary of the certificate information.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /system/certificate
    {
        "status": "successful",
        "type": "csr",
        "data": "-----BEGIN CERTIFICATE REQUEST-----
\nMIIEwzCCAqsCAQAwfjELMAkGA1UEBhMCVXMxEDAOBgNVBAgMB0FsYWJhbWExEzAR\nBgNVBAcMCkh1bnRzdmlsb
GUxDzANBgNVBAoMBlZlcnRpdjEPMA0GA1UECwwGVmVy\ndGl2MSYwJAYJKoZIhvcNAQkBFhdiaWxsLmxhemVuYnlAdm
VydGl2LmNvbTCCAiIw\nDQYJKoZIhvcNAQEBBQADgg...2WByY7bmauvF6WlfaZh7FIVKvhEjSLn+YavGN8\n1ThY2Kz9PTc
MnOAI793IkzzzqgL0jVrQYJmW/4qJaeF7VZ675KQVGKbc5Yi89HBh\npITSC+ou57tRrI1fbdFBPqOrjLq1ueVQJ1pc7HJ9VCA
OogtgH3JFhyRYf12iDrrs\n5YJmJO7LRbvALAo8DFGsOMFQiVL/dfA=\n-----END CERTIFICATE REQUEST-----\n"
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

GET /system/certificate


```
{
    "status": "successful",
    "type": "cert",
    "data": "Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            25:8a:3a:6b:00:f5:1b:fe:e6:9a:6e:0f:81:64:fc:a2:65:18:c6:48
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C = Us, ST = Alabama, L = Huntsville, O = Vertiv, OU = Vertiv, emailAddress = bill.lazenby@vertiv.com
        Validity
            Not Before: Jul 14 18:32:29 2022 GMT
            Not After : Jul 11 18:32:29 2032 GMT
        Subject: C = Us, ST = Alabama, L = Huntsville, O = Vertiv, OU = Vertiv, emailAddress = bill.lazenby@vertiv.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                RSA Public-Key: (4096 bit)"
}
```


GET /system/certificate?format=base64


```
{
    "status": "successful",
    "type": "csr",
    "data":
"LS0tLS1CRUdJTiBDRVJUSUZJQ0FURSBSRVFVRVNULS0tLS0KTUlJRXpUQ0NBclVDQVFBd2dZY3hDekFKQmdOVkJB
WVRBbIZ6TVJBd0RnWURWUVFJREFkaGJHR
    mlZVzFoTVJNdwpFUVIEVlFRSERBcG9kVzUwYzNacGJHeGxNUTh3RFFZRFZRUUtEQVoyWlhKMGFYWXhFREFP ...
1hxMWtQWkNPNXJRNmpsQWtpZUNWNEhuN2xFSW14

cU1jMTBDZkp1UmRpR2t1YWhveUJHRgp3ZGRXV2I3UlJ6dTE0VnNyRHF1SmhPbzh5RUNpNXVhSFIrUGNpL0RyQWdwQ
gotLS0tLUVORCBDRVJUSUZJQ0FURSBSRVFVRVNU
    LS0tLS0K"
}
```


GET /system/certificate?format=base64


```
{
    "status": "in progress",
    "type": "cert",
    "data": ""
}
```

2.2.18 /system/certificate/upload

Use this resource to upload a certificate file contained in the body of the JSON message and checks that it is valid for this appliance.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
data	Base64 encoded data for the certificate.
passphrase	Passphrase if required for an embedded key file. If none is provided, the appliance will attempt to use the current key and pass phrase or the previously retained key and pass phrase from the CSR Generation step.

Response Body

Parameter	Description
status	Status result, typically “certificate upload successful” on success.

Response Codes

200	OK
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
POST /system/certificate/upload
    {
        "data": "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURSBSRVFVRVNULS0tLS0KTUlJRXpU

Q0NBclVDQVFBd2dZY3hDekFKQmdOVkJBWVRBbIZ6TVJBd0RnWURWUVFJREFkaGJHRmlZVzFoTVJNdwpFUVlEVl
FRSERBcG9kVzUwYzNacGJHeGxNUTh3RFFZRF

ZRUUtEQVoyWlhKMGFYWXhFREFP...1hxMWtQWkNPNXJRNmpsQWtpZUNWNEhuN2xFSW14cU1jMTBDZkp1UmRp
R2t1YWhveUJHRgp3ZGRXV2I3UlJ6dTE0VnNyRHF
        1SmhPbzh5RUNpNXVhSFIrUGNpL0RyQWdwQgotLS0tLUVORCBDRVJUSUZJQ0FURSBSRVFVRVNULS0tLS0K"
    }
```

Response is:

```
    {
        "status": "certificate upload successful",
    }
```

2.2.19 /system/certificate/apply

Use this resource to apply the previously downloaded or generated certificate.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “certificate application successful” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /system/certificate/apply
        {
            "status": "certificate application successful",
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.20 /system/dateAndTime

Use this resource to get and set parameters for the system's date and time.

NOTE: When NTP is enabled, setting the time and date is not permitted and results in an error.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
time	Current time.
date	Current date.
ntp	Network Time Protocol: **enabled** / **disabled**.
ntpSettings (only valid when ntp is enabled)	
ntpServer	IP address or name.
ntpServer2	IP address or name.
ntpAuthentication	NTP Server Authentication: **enabled** / **disabled**
ntpKeyId1	Server 1 Key ID.
ntpKeyType1	Server 1 Key Type: **MD5**/ **SHA1**
ntpKey1	Server 1 Key.
ntpKeyId2	Server 2 Key ID.
ntpKeyType2	Server 2 Key Type: **MD5** / **SHA1**
ntpKey2	Server 2 Key.
timezone	Name of the time zone. Default: UTC

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /system/dateAndTime
    {
        "time": "18:00:50",
        "date": "07/12/2017",
        "ntp": "enabled",
        "ntpSettings": {
            "ntpServer": "10.20.30.40",
            "ntpServer2": "ntp.pool.org"
            "ntpAuthentication": "enabled",
            "ntpKeyId1": "11",
            "ntpKeyType1": "SHA1",
            "ntpKey1": "",
            "ntpKeyId2": "1",
            "ntpKeyType2": "MD5",
            "ntpKey2": ""
        },
    "timezone": "UTC"
    }
```

```
PUT /system/dateAndTime
        {
            "ntp": "disabled"
        }
```

```
PUT /system/dateAndTime
        {
            "time": "05:00:00",
            "date": "08/01/2017"
        }
```

```
PUT /system/dateAndTime
        {
            "ntp": "enabled",
            "ntpSettings": {
                "ntpServer": "ntp.pool.org",
                "ntpServer2": "",
                "ntpAuthentication": "enabled",
                "ntpKeyId1": "20",
                "ntpKeyType1": "SHA1",
                "ntpKey1": ""
            }
        }
```

```
PUT /system/dateAndTime {"timezone":"US/Central"}
```

```
PUT /system/dateAndTime {"timezone":"Custom"}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.21 /system/dateAndTime/testNtp

Use this resource to test the current NTP settings.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Status result either “success” or “failure”.
detail	Detailed output of the NTP test script for debugging.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
POST /system/dateAndTime/testNtp
```

Response is:

```
{
    "status": "success",
    "detail": " ntpd: 'pool.ntp.org' is 23.106.249.200\nntpd: sending query to 23.106.249.200\nntpd: reply from 23.106.249.
            200: offset:+0.043805 delay:0.254510 status:0x24 strat:2 refid:0
            x0a018582rootdelay:0.248142 reach:0x01\nntpd: sending query to 23.106.249
            .200\nntpd: reply from 23.106.249.200: offset:+0.047282 delay:0.255508 status:0x24 strat:2 ref
            id:0x0a018582 rootdelay:0.248142 reach:0x03\n\nNTP Server 1 Response Passed.\n"
}
```

2.2.22 /system/dateAndTime/timezones

Use this resource to obtain a list of all recognized time zones. These values can be used when setting the time zone via the /system/dateAndTime resource.

Methods

GET

Parameters

None.

Request Body

None.

Response Body

Array of time zone name strings:

Parameter	Description
name	Name of a timezone.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /system/dateAndTime/timezones
        {
            "timezones": [
                "Africa/Abidjan",
                "Africa/Accra",
                ....
                "Zulu"
            ]
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.23 /system/dateAndTime/timezone/custom

Use this resource to access information and configure the settings for a custom time zone.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
name	Time zone name
acronym	Standard time acronym
gmtOffset	Example: -7:00
dst	Daylight savings time. **enabled** / **disabled**
(Only used when dst is enabled):	
dstAcronym	Daylight Saving Time acronym.
saveTime	Example: 0:00 to 5:00
startMonth	**january**/.../**december**
startWeek	Week of the month (integer): 1-5
startWeekday	**sunday**/.../**saturday**
startHour	Example: 01:00
endMonth	**january**/.../**december**
endWeek	Week of the month (integer): 1-5
endWeekday	**sunday**/.../**saturday**
endHour	Example: 01:00

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /system/dateAndTime/timezone/custom

```
{
    "name": "mycustom",
    "acronym": "myc",
    "gmtOffset": "-4:00",
    "dst": "enabled",
    "dstAcronym": "mydst",
    "saveTime": "1:00",
    "startMonth": "april",
    "startWeek": 1,
    "startWeekday": "sunday",
    "startHour": "02:00",
    "endMonth": "november",
    "endWeek": 1,
    "endWeekday": "sunday",
    "endHour": "02:00",
}
```

PUT /system/dateAndTime/timezone/custom

```
{
    "name": "mycustom",
    "acronym": "myc",
    "gmtOffset": "-4:00",
    "dst": "disabled"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.24 /system/general

Use this resource to get and set general appliance level parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
language	Language used for SSH, Telnet and console port sessions to the appliance: **english** / **chinese** / **french** / **german** / **japanese** / **spanish**
onlineHelp	URL for the online help product documentation. Default: http://global.avocent.com/us/olh/acs8x/en/index.html.
banner	Controls whether a login banner is displayed: **enabled** / **disabled**
bannerText	Text of the login banner to display. It may include special characters for formatting including \t for tabs and \n for newlines.
bannerAck	Controls whether to display a banner acknowledgment checkbox: **enabled** / **disabled**
viewer	Type of viewer to use when opening serial or appliance sessions: **html5** / **jnlp**

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /system/general
    {
        "language": "english",
        "onlineHelp": "http://global.avocent.com/us/olh/acs8x/en/index.html",
        "banner": "enabled",
        "bannerText":
            "==========================================================================\n
            WARNING! The use of this system is restricted to authorized users. \n   \n   All information
            and communications on this system are subject \n   to review, monitoring and recording at any
            time, without notice\n   or permission. Users should have no expectation of privacy.
            \n==========================================================================\n",
        "bannerAck": "disabled",
        "viewer": "html5"
    }
```

```
PUT /system/general {"bannerText":"==========\n Multiple\n   Line\n  Banner\n==========\n"}
```

```
PUT /system/general {"language":"german","banner":"disabled","viewer":"jnlp"}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.25 /system/bootConfig

Use this resource to get and set parameters for the system's boot configuration.

Methods

GET, PUT, PATCH

Parameters

None.

Request Body

Parameter	Description
bootMode	flash / network
bootImage	Image number when booted from flash (integer): 1 / 2
flashImage1	Read-only firmware version and date of the first image in flash.
flashImage2	Read-only firmware version and date of the second image in flash.
applianceIp	Appliance IP address when booted from network. Example: 10.20.30.40
tftpServerIp	TFTP Server IP address when booted from network. Example: 10.20.30.41
filename	Filename when booted from network.
watchdog	Watchdog timer to reboot the system in the event of a crash: enabled / disabled
consoleSpeed	Console port speed (integer): 1200 / 2400 / 4800 / 9600 / 19200 / 38400 / 57600 / 115200

Response Body

Same as request body.

Response Codes

200	OK
400	Bad Request

Examples

GET /system/bootConfig

```
{
    "bootFrom": "flash",
    "bootImage": 1,
    "flashImage1": "1.3.12.2683+551+26+1 (Jun 15 2017 - 16:26:07)",
    "flashImage2": "1.3.75.2779+551+28+11 (Jul 26 2017 - 10:02:10)",
    "watchdog": "enabled",
    "consoleSpeed": 9600
}
```

GET /system/bootConfig

```
{
    "bootFrom": "network",
    "applianceIp": "10.20.30.40",
    "tftpServerIp": "10.20.30.41",
    "filename": "acs.fl",
    "watchdog": "enabled",
    "consoleSpeed": 9600
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.26 /system/usage/memory

Use this resource to get read-only system information about memory usage.

Methods

GET

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
active	Memory that has been used more recently and usually not reclaimed unless necessary. Example: "66856 kB"
anonPages	Non-file backed pages mapped into userspace page tables.
bounce	Memory used for block device "bounce buffers."
buffers	Relatively temporary storage for raw disk blocks.
cached	In-memory cache for files read from the disk.
commitLimit	Total amount of memory currently available to be allocated on the system.
committedAS	Amount of memory presently allocated on the system.
dirty	Memory which is waiting to get written back to the disk.
highFree	Highmem is all memory above ~860MB of physical memory; Highmem areas are for use by userspace programs, or for the pagecache.
highTotal	Highmem total.
inactive	Memory which has been less recently used.
kernelStack	Memory the kernel stack uses.
lowFree	Memory used for everything that highmem can be used for; also available for the kernel's use for its own data structures.
lowTotal	Lowmem total.
mapped	Files which have been mapped, such as libraries.
memAvailable	Estimate of how much memory is available for new applications, without swapping.
memFree	Sum of LowFree+HighFree.
memTotal	Total usable ram.
mlocked	Pages locked to memory using the mlock( ) system call.

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
nfsUnstable	NFS pages sent to the server, but not yet committed to stable storage.
pageTables	Amount of memory dedicated to the lowest level of page tables.
shmem	Total memory used by shared memory (shmem) and tmpfs.
slab	In-kernel data structures cache.
sReclaimable	Part of Slab, that might be reclaimed, such as caches.
sUnreclaim	Part of Slab, that cannot be reclaimed on memory pressure.
swapCached	Memory present within main memory, but also in the swapfile.
swapFree	Memory evicted from RAM, and temporarily on the disk.
swapTotal	Total amount of swap space available.
unevictable	Unevictable pages can't be swapped out for a variety of reasons.
vmallocChunk	Largest contiguous block of vmalloc area which is free.
vmallocTotal	Total size of vmalloc memory area.
vmallocUsed	Amount of vmalloc area used.
writeback	Memory being actively written back to the disk.
writebackTmp	Memory used by FUSE for temporary writeback buffers.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /system/usage/memory
        {
            "active": "66856 kB",
            "activeAnon": "38180 kB",
            ...
            "writebackTmp": "0 kB"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.2.27 /system/usage/flash

Use this resource to get read-only system information about flash usage and return an array of file systems and their individual usage information.

Methods

GET

Parameters

Fields and filtering queries are supported for all body parameters.

Request Body

None.

Response Body

Array of file system information:

Parameter	Description
filesystem	Name of the file system.
1kBlocks	Total number of 1k blocks.
used	Number of blocks used.
available	Number of block available.
use	Percent of blocks in use.
mountedOn	Directory path where the file system is mounted.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /system/usage/flash
        {
            "flashUsage":   [
                {
                    "filesystem": "/dev/mmcblk0p8",
                    "1kBlocks": "59361",
                    …
                },
                {
                    "filesystem": "/dev/mmcblk0p11",
                    …
                }
            ]
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.3 Security Profile

2.3.1 /security

Use this resource to get and set parameters in the security profile of the appliance.

NOTE: API changes made while using the API may terminate the API session.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
idleTimeout	Session idle timeout in seconds (integer). 0 disables the timeout entirely. Minimum accepted value: 90.
rpc	RPC service: **enabled** / **disabled**
pluggableDevices	Pluggable device detection: **enabled** / **disabled** **NOTE: Changes made to Pluggable Devices, including disablement, are only effective after reboot.**
pluggableStorage	Pluggable storage devices: **enabled** / **disabled**
oneWire	OneWire devices: **enabled** / **disabled**
access	Serial port access can be configured to allow access for all users, or allow the configuration of group and user-specific authorizations to restrict access: **all** / **user_group**
allAccessSettings (only valid when access is set to "all")	
session	Sessions can be set to allow single or multiple session read/write: **single** / **multiple**
killMultiSession	Enables you to kill other sharers of a session: **enabled** / **disabled**
sendMessageMultiSession	Enables you to send a message to other users sharing a session: **enabled** / **disabled**
powerControl	Enables you to control power of a multi-session: **enabled** / **disabled**
dataBufferManagement	Enables you for data buffer management of a session: **enabled** / **disabled**
restfulClientMenu	Enables the restfulClient menu: **enabled** / **disabled**
eraseFlash	Control whether entire flash configuration is erased on a factory default: **enabled** / **disabled**
bootp	Controls bootp configuration retrieval: **enabled** / **disabled**
bootpSettings (only valid when bootp is enabled)	
bootpInterface	Specifies network interface used by bootp: **eth#**
liveConfigurationRetrieval	Enables live configuration retrieval any time DHCP renews: **enabled** / **disabled**
sshUserPass	Controls whether SSH allows authentication via username/password: **enabled** / **disabled**

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
profile	Sets the appliance security profile: **open** / **moderate** / **secure** / **custom**
customProfile (only valid when profile is set to "custom")	
telnet	Telnet service: **enabled** / **disabled**
ftp	FTP service: **enabled** / **disabled**
tftp	TFTP service: **enabled** / **disabled**
tftpRoot	TFTP root directory when tftp is enabled. Example: /mnt/hdUser
snmp	SNMP service: **enabled** / **disabled**
ipsec	IPSec: **enabled** / **disabled**
answerIcmp	Answer ICMP message: **enabled** / **disabled**
sshVersion	SSH Version: **2**
sshPort	SSH TCP port number (integer). Default: 22
sshRootAccess	SSH allow root access: **enabled** / **disabled**
sshCipherLevel	SSH minimum cipher and mac suite level: **low** / **high** / **custom**
sshCustomCipher	SSH custom Ciphers.
sshCustomMac	SSH custom MACs.
sshCustomKex	SSH custom KexAlgorithms.
sshCustomHostKey	SSH custom HostKeyAlgorithms.
httpSession	HTTP sessions: **enabled** / **disabled** **NOTE: Profile changes affecting HTTP and HTTPS terminate all http sessions.**
httpSettings (only valid when httpSession is enabled)	
httpPort	HTTP port number (integer). Default: 80
httpsSession	HTTPS sessions: **enabled** / **disabled**
httpsSettings (only valid when httpsSession is enabled)	
httpsTlsVersion	HTTPS TLS minimum version: **1.0** / **1.1** / **1.2** / **1.3**
httpsCipherLevel	HTTPS minimum cipher suite level: **low** / **medium** / **high** / **custom**
httpsCustomCipherList	TLS custom cipher suite list. These do not include the TLS 1.3 cipher as those are handled separately. **NOTE: Only applicable when httpsCipherLevel is set to "custom".**
httpsCustomTls13CipherList	TLS 1.3 specific custom cipher suite list. **NOTE: Only applicable when httpsCipherLevel is set to "custom".**
httpsPort	HTTPS Port number (integer). Default: 443
redirectHttp	Redirect HTTP/HTTPS: **enabled** / **disabled**
httpSTS	HTTP Strict Transport Security: **enabled** / **disabled**

Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
consolePort	Appliance console port: **enabled** / **disabled** **NOTE: Disabling the Console Port can make the appliance inaccessible and should only be done in the most extreme cases.**
apiHttpAccess	Allow RESTful API access via HTTP: **enabled** / **disabled**
apiHttpSettings (only valid when apiHttpAccess is enabled)	
apiHttpPort	HTTP port number for RESTful API access (integer). Default: 8080
apiHttpsAccess	Allows RESTful API access via HTTPS: **enabled** / **disabled**
apiHttpsSettings (only valid when apiHttpsAccess is enabled)	
apiHttpsPort	HTTPS port number for RESTful API access (integer). Default: 48048
fips	FIPS 140-2 module: **enabled** / **disabled** **NOTE: Changing the fips value causes the appliance to reboot.**
dsview	Allows appliance to be managed by Vertiv™ management software: **enabled** / **disabled**
dsviewCertificate	Certificate for enrollment, bit size. **1024** / **2048**
managerType	Read-only. Current management type: **adx** / **dsview** / **unmanaged** **NOTE: Only reports "dsview" if appliance is enrolled into the Vertiv™ Avocent® DSView™ 4.5 management software in Secure mode.**
managerIpAddress	Read-only. IP address of the current management appliance.

Response Body

Same as response body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /security
        {
                "idleTimeout": 63955,
                "rpc": "disabled",
                "pluggableDevices": "enabled",
                 "pluggableStorage": "disabled",
                "oneWire": "enabled"<
                "access": "all",
                "allAccessSettings": {
```

```
        "session": "multiple",
        "killMultiSession": "enabled",
        "sendMessageMultiSession": "enabled",
        "powerControl": "enabled",
        "dataBufferManagement": "enabled",
        "restfulClientMenu": "enabled"
    },
    "eraseFlash": "disabled",
    "bootp": "enabled",
    "bootpSettings": {
        "bootpInterface": "eth0",
        "liveConfigurationRetrieval": "enabled"
    },
    "sshUserPass": "enabled",
    "profile": "custom",
    "customProfile": {
        "telnet": "enabled",
        "ftp": "disabled",
        "tftp": "disabled",
        "snmp": "disabled",
        "ipsec": "disabled",
        "answerIcmp": "enabled",
        "sshVersion": "2",
        "sshPort": 22,
        "sshRootAccess": "enabled",
        "sshCipherLevel": "low",
        "sshCustomCipher": "aes128-ctr,aes192-ctr,aes256-ctr",
        "sshCustomMac": "hmac-sha2-256-etm@openssh.com",
        "sshCustomKex": "-diffie-hellman-group14-sha1",
        "sshCustomHostKey": "rsa-sha2-256,rsa-sha2-512,ssh-ed25519”,
        "httpSession": "enabled",
        "httpSettings": {
            "httpPort": 80
        },
        "httpsSession": "enabled",
        "httpsSettings": {
            "httpsTlsVersion": "1.1",
            "httpsCipherLevel": "low",
            "httpsCustomCipherList": "ECDHE-RSA-AES128-GCM-SHA256",
            "httpsTls13CustomCipherList": "TLS_AES_256_GCM_SHA384",
            "httpsPort": 443,
            "redirectHttp": "disabled",
            "httpSTS": "disabled"
        }
    },
    "consolePort": "enabled",
    "apiHttpAccess": "enabled",
    "apiHttpSettings": {
        "apiHttpPort": 8080
    },
    "apiHttpsAccess": "enabled",
    "apiHttpsSettings": {
        "apiHttpsPort": 48048
    },
    "fips": "disabled",
    "dsview": "enabled",
    "dsviewCertificate": "1024",
    "managerType": "dsview",
```


 Proprietary and Confidential ©2025 Vertiv Group Corp.

```
        "managerIpAddress": "10.20.30.40"
    }
```

GET /security

```
    {
        "idleTimeout": 63955,
        "rpc": "disabled",
        "pluggableDevices": "disabled",
        "access": "user_group",
        "eraseFlash": "disabled",
        "bootp": "disabled",
        "sshUserPass": "enabled",
        "profile": "open",
        "consolePort": "enabled",
        "apiHttpAccess": "disabled",
        "apiHttpsAccess": "disabled",
        "fips": "disabled",
        "dsview": "disabled"
        "dsviewCertificate": "1024"
    }
```

GET /security?fields=idleTimeout

```
    {
        "idleTimeout": 300
    }
```

GET /security?fields=rpc,pluggableDevices

```
    {
        "rpc": "disabled",
        "pluggableDevices": "enabled"
    }
```

PUT /security {"customProfile": {"httpsSettings": {"httpsPort": 444, "redirectHttp": "disabled"}}}

2.3.2 /security/clearDSView

Use this resource to clear the certificates installed by Vertiv™ management software on the appliance.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “certificate cleared” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /security/clearDSView
    {
        "status": "certificate cleared"
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4 Network

2.4.1 /network/settings

Use this resource to get and set various network parameter settings of the console system.

NOTE: When configuring failoverSettings, the primary and secondary interface cannot be set to the same value. Therefore, when swapping them, the changes need to be made in a single PUT to the resource.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
hostname	User-defined name of the appliance on the network. Default format: ACS80xx-<serialNumber>
primaryDns	IP address of the primary DNS server.
secondaryDns	IP address of the secondary DNS server.
domain	Domain address. Default: corp.avocent.com
search	Search address. Default: corp.avocent.com
lldp	Link Layer Discovery Protocol: **enabled** / **disabled**
ipv6	IPv6 support: **enabled** / **disabled**
ipv6Settings (only valid when ipv6 is enabled)	
dhcpv6Dns	Get the IPv6 DNS server address from DHCPv6: **enabled** / **disabled**
dhcpv6Domain	Get the IPv6 Domain name from DHCPv6: **enabled** / **disabled**
dhcpv6Bootfile	Get the ZTP bootfile URL from DHCPv6: **enabled** / **disabled**
ipsecFrequency	IPSec Tunnel Check Frequency.
ipsecMaxTime	IPSec Tunnel Check Maximum Time.
multipleRouting	Enable network failover or IPv4 Multiple Routing Tables: **none** / **enable_network_failover** / **enable_multiple_routing**
bonding	Bonding with eth0 as primary and eth1 as second mode of access: **enabled** / **disabled**
bondingSettings: (only valid when bonding is enabled)	
miiMon	Milliseconds (integer).
upDelay	Milliseconds (integer).

Parameter	Description
failoverSettings (only valid when multipleRouting is set to "enable_network_failover").	
primaryInterface	Primary network interface: eth# / bond0
secondaryInterface	Secondary network interface: eth# / lte
vpnConnection	Name of an existing VPN connection.
trigger	Failover trigger: primary_interface_down / unreachable_primary_default_gateway / unreachable_ip_address / unreachable_dsview
unreachableIp	IP address to probe when trigger is set to "unreachable_ip_address".
useCustomIp	Provide a custom IP address to the Vertiv™ management software to use during failover.
customIp	IP address to provide to the Vertiv™ management software to use during failover.
multiRouteSettings: (only valid when multipleRouting is set to “enable_multiple_routing”)	
additionalInterface	Additional interface to be included in multiple routing in addition to eth0 and eth1: none / lte / eth#

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /network/settings

```
{
    "hostname": "ACS8048-123456789",
    "primaryDns": "10.20.30.40",
    "secondaryDns": "10.20.30.50",
    "domain": "corp.avocent.com",
    "search": "corp.avocent.com",
    "lldp": "disabled",
    "ipv6": "enabled",
    "ipv6Settings": {
        "dhcpv6Dns": "disabled",
        "dhcpv6Domain": "disabled",
    }
    "ipsecFrequency": "1m",
    "ipsecMaxTime": "24h",
    "multipleRouting": "enable_network_failover",
    "failoverSettings": {
        "primaryInterface": "eth0",
        "secondaryInterface": "eth1",
        "vpnConnection": "",
        "trigger": "unreachable_ip_address",
        "unreachableIp": "10.20.30.50",
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
                "useCustomIp": “disabled”
            },
            "bonding": "disabled"
        }
```

PUT /network/settings {"hostname": "myhostname"}

2.4.2 /network/devices[/<INTERFACE>]

Use this resource to get and set various network device parameters for individual network interfaces.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Parameter	Description
INTERFACE	Name of the network interface: eth0, eth1, bond0, etc.

Request Body

Parameter	Description
interface	Read-only name of the network interface: eth0, eth1 and so on.
isPrimary	Identifies if this is the primary interface: **enabled** / **disabled**. To change this value, the network interface that should be primary has this field set to enabled. All other interfaces will then report disabled.
status	Status of the interface: **enabled** / **disabled**
ipv4Method	Method used to configure IPv4: **dhcp** / **static** / **unconfigured**
ipv6Method	Method used to configure IPv6: **stateless** / **dhcpv6** / **static** / **unconfigured**
mac	Read-only hardware MAC address of this interface. Example: 00:e0:86:01:02:03
ipv4Static (only valid when ipv4Method is set to "static")	
ipv4Address	IPv4 address for static configuration. Example: 10.20.30.40
ipv4Mask	Subnet mask for static configuration. Example: 255.255.255.0
ipv4Gateway	Gateway to use for static configuration. Example: 10.20.30.1
ipv6Static (only valid when ipv6Method is set to "static")	
ipv6Address	IPv6 address for static configuration. Example: fd00:0024:0000:0000:02e0:86ff:fe10:2c3b
ipv6PrefixLength	IPv6 prefix length for static configuration. Example: 112
mtu	Maximum Transmission Unit size in bytes (integer).
mode	Auto negotiation mode: **auto** / **noauto** **NOTE: Noauto is only supported on eth0 and eth1 and should NOT be used for copper connections. Noauto should only be used for fiber connections when absolutely necessary.**

Proprietary and Confidential ©2025 Vertiv Group Corp.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /network/devices

```
{
    "devices": [
        {
            "interface": "eth0",
            "isPrimary": "enabled",
            "status": "enabled",
            "ipv4Method": "dhcp",
            "ipv6Method": "stateless",
            "mac": "00:11:22:33:44:55",
            "mtu": 1500
        },
        {
            "interface": "eth1",
            …
        }
    ]
}
```

GET /network/devices/eth1

```
{
    "interface": "eth1",
    "isPrimary": "disabled",
    "status": "enabled",
    "ipv4Method": "static",
    "ipv6Method": "stateless"
    "mac": "00:11:22:33:44:56",
    "ipv4Static": {
        "ipv4Address": "10.20.30.40",
        "ipv4Mask": "255.255.255.0",
        "ipv4Gateway": ""
    }
    "mtu": 1500,
    "mode": "noauto"
}
```

 Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.3 /network/hosts[/<IPADDRESS>]

Use this resource to get and set network hosts entries for the local network.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
IPADDRESS	IP address of the network host.

Request Body

Parameter	Description
ipAddress	IPv4 or IPv6 address for the host. Read-only on a PUT/PATCH.
hostname	Hostname.
alias	Alternate hostname or alias.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /network/hosts

```
{
    "networkHosts": [
        {
            "ipAddress": "127.0.0.1",
            "hostname": "localhost.localdomain",
            "alias": "localhost"
        },
        {
            "ipAddress": "::1",
            "hostname": "localhost",
            "alias": "ip6-localhost ip6-loopback"
        },
        …
    ]
}
```

GET /network/hosts/10.20.30.40

```
{
    "ipAddress": "10.20.30.40",
    "hostname": "myserver",
    "alias": ""
}
```

GET /network/hosts/fe00::0

```
{
    "ipAddress": "fe00::0",
    "hostname": "ip6-mcastprefix",
    "alias": ""
}
```

POST /network/hosts

```
{
    "ipAddress": "10.20.30.70",
    "hostname": "anotherhost.domain",
    "alias": "anotherhost"
}
```

Response is:

```
{
    "ipAddress": "10.20.30.70",
    "hostname": "anotherhost.domain",
    "alias": "anotherhost"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
PUT /network/hosts/10.20.30.70
        {
            "hostname": "newhost.domain",
            "alias": "newhost",
        }
```

```
DELETE /network/hosts/10.20.30.70
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.4 /network/staticRoutes/<TABLE>[/<ID>]

Use this resource to get and set static route table entries.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
TABLE	Specifies the IP table: **ipv4** or **ipv6**
ID	Identifier of an existing entry composed of the destination and metric fields separated by a hyphen. Example: default-3

Request Body

Parameter	Description
id	Read-only identifier composed of destination and metric fields.
destination	Destination IP / Mask bits or "default". Example: "10.20.30.40/32"
gateway	Gateway IP address.
interface	Interface to which the route applies: **eth#** / **lte0**
metric	Metric value between 0 and 32765 (integer).

Response Body

Same as request body.

Response Codes

200	OK
201	Created
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /network/staticRoutes/ipv4

```
{
    "staticRoutes": [
        {
            "id": "10.20.30.40/32-123",
            "destination": "10.20.30.40/32",
            "gateway": "10.20.30.41",
            "interface": "eth0",
            "metric": 123
        },
        {
            "id": "default-3",
            "destination": "default",
            "gateway": "1.2.3.4",
            "interface": "eth0",
            "metric": 3
        }
    ]
}
```

GET /network/staticRoutes/ipv4/default-3

```
{
    "id": "default-3",
    "destination": "default",
    "gateway": "1.2.3.4",
    "interface": "eth0",
    "metric": 3
}
```

DELETE /network/staticRoutes/ipv4/default-3

POST /network/staticRoutes/ipv4

```
{
    "destination": "10.20.30.40/32",
    "gateway": "1.2.3.4",
    "interface": "eth1",
    "metric": 123
}
```

Response is:

```
{
    "id": "10.20.30.40/32-123",
    "destination": "10.20.30.40/32",
    "gateway": "",
    "interface": "eth1",
    "metric": 123
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.5 /network/firewall/<TABLE>[/<CHAIN>]

Use this resource to create and configure chains for the IPv4 and IPv6 filter tables.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
TABLE	Specifies the IP table: **ipv4** or **ipv6**
CHAIN	Specifies the chain name to access: **FORWARD** / **INPUT** / **OUTPUT** / *****

Request Body

Parameter	Description
name	Chain name: **FORWARD** / **INPUT** / **OUTPUT** / *****
policy	Chain policy: **accept** / **drop**
packets	Number of packets processed for built-in chains only (integer).
bytes	Number of bytes processed for built-in chains only (integer).

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /network/firewall/ipv4

```
{
    "chains": [
        {
            "name": "FORWARD",
            "policy": "accept",
            "packets": 0,
            "bytes": 0,
        },
        {
            "name": "INPUT",
            "policy": "accept",
```

```
                "packets": 124294,
                "bytes": 15373123,
            },
            {
                "name": "INPUT",
                …
            }
        ]
    }
```

GET /network/firewall/ipv6/FORWARD

```
    {
        "name": "FORWARD",
        "policy": "accept","packets": 0,
        "bytes": 0,
    }
```

POST /network/firewall/ipv4

```
    {
        "name":"mychain"
    }
```

Response is:

```
{
    "name": "mychain",
    "policy": "-",
    "packets": 0,
    "bytes": 0,
}
```

DELETE /network/firewall/ipv4/mychain

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.6 /network/firewall/<TABLE>/<CHAIN>/rules[/<INDEX>]

Use this resource to configure rules for a firewall chain.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
TABLE	Specifies the IP table: **ipv4** or **ipv6**
CHAIN	Specifies the chain name to access: **FORWARD** / **INPUT** / **OUTPUT** / *****
INDEX	Specifies the integer index of the existing rule to access.

Request Body

Parameters	Description
index	Rule index assigned by system for identification (integer).
target	**accept** / **drop** / **log** / **reject** / **return**
source	Source address and mask or prefix. Simple IP or Classeless Inter-Domain Routing (CIDR) or network with dotted quad netmask.
destination	Destination address and mask or prefix. Simple IP or CIDR or network with dotted quad netmask.
protocol	**tcp** / **udp** / **icmp** / **icmpv6** / **all** or a number in string form with a value between 0 and 255.
(Only used when protocol is set to "tcp"):	
tcpSourcePort	Port number, blank, or a colon separated range of port numbers.
tcpDestPort	Port number, blank, or a colon separated range of port numbers.
tcpFlagSyn	**any** / **set** / **unset**
tcpFlagAck	**any** / **set** / **unset**
tcpFlagFin	**any** / **set** / **unset**
tcpFlagRst	**any** / **set** / **unset**
tcpFlagUrg	**any** / **set** / **unset**
tcpFlagPsh	**any** / **set** / **unset**
(Only used when protocol is set to "udp"):	
udpSourcePort	Port number, blank or a colon separated range of port numbers.
udpDestPort	Port number, blank, or a colon separated range of port numbers.
(Only used when protocol is set to “icmp” or “icmpv6”):	
icmpType	One of the following:

Parameters	Description
	address_mask_reply / address_mask_request / any / bad_ip_header / communication_prohibited / destination_unreachable / echo_reply / echo_request / fragmentation_needed / host_precedence_ violation / host_prohibited / host_redirect / host_unknown / host_unreachable / network_prohibited / network_redirect / network_unknown / network_unreachable / parameter_problem / port_unreachable / precedence_cutoff / protocol_unreachable / redirect / required_option_missing / router_advertisement / router_solicitation / source_quench / source_route_failed / time_exceeded / timestamp_reply / timestamp_request / tos_host_redirect / tos_host_unreachable / tos_network_redirect / tos_network_ unreachable / ttl_zero_during_reassembly / ttl_zero_during_transit
inputInterface	any / lo / eth# / lte0
outputInterface	any / lo / eth# / lte0
fragments	all_packets_and_fragments / unfragmented_packets_and_1st_packets / 2nd_and_further_packets
matchSourceIpDifferent	enabled / disabled
matchDestIpDifferent	enabled / disabled
matchSourcePortsDifferent	enabled / disabled
matchDestPortsDifferent	enabled / disabled
matchProtocolsDifferent	enabled / disabled
matchTcpFlagsDifferent	enabled / disabled
matchIcmpTypesDifferent	enabled / disabled
matchInputInterfacesDifferent	enabled / disabled
matchOutputInterfacesDifferent	enabled / disabled
(Only used when target is set to "log"):	
logLevel	debug / info / notice / warning / error
logPrefix	Prefix to use in log entry.
(Only used when target is set to "reject"):	
rejectWith	administratively_prohibited / host_ prohibited / host_unreachable / network_ prohibited / network_ unreachable / port_unreachable / protocol_unreachable / tcp_reset
packets	Number of packets processed (integer).
bytes	Number of bytes processed (integer).

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /network/firewall/ipv4/FORWARD/rules

```
{
    "rules": [
        {
            "index": 0,
            "target": "log",
            "sourceIpMask": "",
            …
        },
        {
            "index": 1,
            …
        }
    ]
}
```

GET /network/firewall/ipv4/FORWARD/rules/1

```
{
    "index": 1,
    "target": "log",
    "sourceIpMask": "10.20.0.0/16",
    "destIpMask": "10.20.30.0/24",
    "protocol": "123",
    "inputInterface": "any",
    "outputInterface": "any",
    "fragments": "all_packets_and_fragments",
    "matchSourceDifferent": "disabled",
    …
    "logLevel": "debug",
    "logPrefix": "myPrefix"
}
```

POST /network/firewall/ipv4/mychain/rules

```
{
    "target":"accept",
    "sourceIpMask": "10.20.0.0/16",
    "destIpMask": "10.20.30.0/24",
    "protocol": "123",
    …
}
```

Response is:

```
{
    "index": 0,
    "target": "accept",
    "sourceIpMask": "10.20.0.0/16",
    "destIpMask": "10.20.30.0/24",
    "protocol": "123",
    "inputInterface": "any",
    "outputInterface": "any",
```

 Proprietary and Confidential ©2025 Vertiv Group Corp.

```
        "fragments": "all_packets_and_fragments",
        "matchSourceDifferent": "disabled",
        …
    }
```

```
DELETE /network/firewall/ipv4/mychain/rules/0
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.7 /network/firewall/<TABLE>/<CHAIN>/rules/<INDEX>/move

Use this resource to move a firewall rule up or down the list or to a specific location in the list of rules.

Methods

POST

Parameters

Parameter	Description
TABLE	Specifies the IP table: **ipv4** or **ipv6**
CHAIN	Specifies the chain name to access: **FORWARD** / **INPUT** / **OUTPUT** / *****
INDEX	Specifies the integer index of the existing rule to access.

Request Body

Parameter	Description
direction	Move up or down by one: **up**/**down**
index	Move to the desired numeric index and shift the other rules accordingly (integer): 0-n where n is one less than the number of rules.

Response Body

None.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
POST /network/firewall/ipv4/INPUT/rules/1/move {"direction": "down"}
```

```
POST /network/firewall/ipv6/INPUT/rules/3/move {"index": 0}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.8 /network/ipsec/connections[/<NAME>]

Use this resource to configure IPSec connections and settings.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Parameter	Description
NAME	Specifies the name of the IPSec connection to access.

Fields query is supported for all body parameters.

Request Body

Parameter	Description
name	IPSec connection name.
tunnelStatus	Read-only current tunnel status: connecting / down / established
ikeVersion	IKE version: **ike_v1** / **ike_v2**
bootAction	Action to be taken when IPSec starts up: **add** / **start** / **ignore** / **route**
aggressive	Whether to use IKEv1 Aggressive or Main Mode: **yes** / **no**
dpdAction	Controls Dead Peer Detection protocol: **none** / **restart**
remoteId	Remote side's ID used for authentication.
remoteIpAddress	IP address of the remote side.
remoteSubnet	Remote side's subnet specified as network/netmask. Example:"10.20.30.00/24"
localId	The console system's ID used for authentication. If blank, the localIpAddress will be used for identification. The localId must be non-blank to differentiate connections if multiple connections have the same localIpAddress.
localVirtualIp	IP address of the console system's interface within the tunnel.
localIpAddress	IP address of the interface that the console system will use for the tunnel.
localSubnet	Local subnet specified as network/netmask. Example: "10.20.30.00/24"
authenticationMethod	Authentication method: **pre_shared_secret** / **psk_and_xauth** / **rsa_certificate**
rsaCertificateFile	RSA certificate filename of a PKCS12 file already on the console system.
rsaFallback	RSA Fallback: **enabled** / **disabled**
xauthPreSharedSecret	XAuth secret used to authenticate to the remote side of the tunnel.
xauthUsername	XAuth username used to authenticate to the remote side of the tunnel.
xauthPassword	XAuth password used to authenticate to the remote side of the tunnel.

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
preSharedSecret	Secret used to authenticate to the remote side of the tunnel.
ikeEncryption	IKE encryption algorithm: 3des / aes128 / aes192 / aes256 / des
ikeHash	IKE integrity or hash algorithm: sha256 / sha384 / sha512 / sha1 / md5
ikeDhGroup	IKE Diffie Hellman Group: modp768 / modp1024 / modp1536 / modp2048 / modp3072 / modp4096 / modp6144 / modp8192 / modp1024s160/ modp2048s224 / modp2048s256 / ecp192 / ecp224 / ecp256 / ecp384 / ecp521 / ecp224bp / ecp256bp / ecp384bp / ecp512bp
ikeStrict	Whether to restrict a responder to specific IKE cipher suites: yes / no
espEncryption	ESP encryption algorithm: 3des / aes128 / aes192 / aes256 / des
espHash	ESP integrity or hash algorithm: sha256 / sha384 / sha512 / sha1 / md5
espDhGroup	ESP Diffie Hellman Group: modp768 / modp1024 / modp1536 / modp2048 / modp3072 / modp4096 / modp6144 / modp8192 / modp1024s160/ modp2048s224 / modp2048s256 / ecp192 / ecp224 / ecp256 / ecp384 / ecp521 / ecp224bp / ecp256bp / ecp384bp / ecp512bp
espStrict	Whether to restrict a responder to specific ESP cipher suites: yes / no
reauthentication	Whether rekeying should also reauthenticate the peer: yes / no
ikeLifetime	Lifetime of the keying channel of the connection. Default: 24h
keyLifetime	Lifetime of the connection. Default: 60m
rekey	Whether to renegotiate the connection when it is about to expire: yes / no
keyingTries	Number of attempts to negotiate a connection before giving up. Default: 3
rekeyMargin	Amount of time before connection expiration to begin renegotiation. Default: 3m
dpdDelay	Delay between messages exchanged as part of Dead Peer Detection protocol. Default: 300s

Response Body

Same as request body.

Response Codes

200	OK
201	Created
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /network/ipsec/connections


```
{
    "connections": [
        {
            "name": "myconnection",
            "tunnelStatus": "down",
            "ikeVersion": "ike_v2",
            "bootAction": "add",
            "aggressive": "no",
            "dpdAction": "none",
            "remoteId": "",
            "remoteIpAddress": "",
            "remoteSubnet": "",
            …
        },
        {
            "name": "anotherconnection",
            "tunnelStatus": "down",
            "ikeVersion": "ike_v1",
            "bootAction": "add",
            "aggressive": "no",
            …
        },
        …
    ]
}
```


GET /network/ipsec/connections/myconnection


```
{
    "name": "myconnection",
    "tunnelStatus": "down",
    "ikeVersion": "ike_v2",
    "bootAction": "add",
    "aggressive": "no",
    "dpdAction": "none",
    "remoteId": "",
    "remoteIpAddress": "10.20.30.40",
    "remoteSubnet": "",
    "localId": "",
    "localVirtualIp": "",
    "localIpAddress": "10.20.30.41",
    "localSubnet": "",
    "authenticationMethod": "pre_shared_secret",
    "rsaCertificateFile": "",
    "rsaFallback": "disabled",
    "xauthPreSharedSecret": "",
    "xauthUsername": "",
    "xauthPassword": "",
    "preSharedSecret": "",
    "ikeEncryption": "aes256",
    "ikeHash": "sha512",
    "ikeDhGroup": "modp768",
    "ikeStrict": "no",
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
        "espEncryption": "aes256",
        "espHash": "sha512",
        "espDhGroup": "modp2048",
        "espStrict": "no",
        "reauthentication": "no",
        "ikeLifetime": "23h",
        "keyLifetime": "60m",
        "rekey": "yes",
        "keyingTries": "3",
        "rekeyMargin": "3m",
        "dpdDelay": "300s"
    }
```

```
DELETE /network/ipsec/connections/myconnection
```

```
PUT /network/ipsec/connections/myconnection
    {
        "remoteId": "mine",
        "remoteIpAddress": "10.20.30.40",
        "remoteSubnet": "10.20.30.00/24"
    }
```

```
POST /network/ipsec/connections
    {
        "name": "myconnection",
        "remoteId": "mine",
        "remoteIpAddress": "10.20.30.40",
        "remoteSubnet": "10.20.30.00/24"
    }
```

Response is:

```
    {
        "name": "myconnection",
        "tunnelStatus": "down",
        "ikeVersion": "ikev2",
        "bootAction": "ignore",
        "aggressive": "no",
        "dpdAction": "none",
        "remoteId": "mine",
        "remoteIpAddress": "10.20.30.40",
        "remoteSubnet": "10.20.30.00/24",
        …
    }
```

2.4.9 /network/ipsec/connections/<NAME>/connect

Use this resource to establish an IPSec connection.

Methods

POST

Parameters

Parameter	Description
NAME	Specifies the name of the IPSec connection to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “initiated connect of <NAME>.” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /network/ipsec/connections/myconnection/connect
```

Response is:

```
{
    "status": "initiated connect of myconnection."
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.10 /network/ipsec/connections/<NAME>/disconnect

Use this resource to disconnect an established IPSec connection.

Methods

POST

Parameters

Parameter	Description
NAME	Specifies the name of the IPSec connection to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “initiated disconnect of <NAME>.” on success.

Examples

```
POST /network/ipsec/connections/myconnection/disconnect
```

Response is:

```
{
    "status": "initiated disconnect of myconnection."
}
```

2.4.11 /network/ipsec/connections/<NAME>/ping

Use this resource to test an established IPSec connection using a network ping.

Methods

POST

Parameters

Parameter	Description
NAME	Specifies the name of the IPSec connection to access.

Request Body

Parameter	Description
ipAddress	The IP address to attempt to ping on the other side of the IPSec connection.
count	Number of pings to send (integer): 1 - 10 Default: 3

Response Body

Parameter	Description
status	Status result, typically “success” on success.
detail	Detailed ping output.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /network/ipsec/connections/myconnection/ping
    {
        "ipAddress": "192.168.1.99",
        "count": 3
    }
```

Response is:

```
    {
        "status": "success",
        "detail": "PING 192.168.7.99 (192.168.7.99) 56(84) bytes of data.\n64 bytes from 192.168.7.99: icmp_seq=1 ttl=255 time
=112 ms\n\n--- 192.168.7.99 ping statistics ---\n1 packets transmitted, 1
```

 Proprietary and Confidential ©2025 Vertiv Group Corp.

```
        received, 0% packet loss, time 0ms\nrtt min/avg/max/mdev = 112.128/112.128/112.128/0.000 ms\n"
}
```

2.4.12 /network/ipsec/connections/<NAME>/diagnostics

Use this resource to view the diagnostic information of the specified IPSec connection.

Methods

GET

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Specifies the name of the IPSec connection to access.

Request Body

None.

Response Body

Parameter	Description
name	IPSec connection name.
tunnelStatus	Current tunnel status: connecting / down / established
localIpAddress	IP address of the interface that the console system will use for the tunnel.
localVirtualIpAddress	IP address of the console system interface within the tunnel.
remoteIpAddress	IP address of the remote side.
remoteSubnet	Subnet IP address and mask of the remote side.
establishedTime	How long the tunnel has been established.
lifetime	Connection's lifetime.
phase1Algorithm	Algorithm negotiated during phase 1.
phase2Algorithm	Algorithm negotiated during phase 1.
certificateName	Name of the certificate use for authentication if appropriate.
connectionStatus	Status output detail from the connection.
lastResult	Result output from the last attempt to establish the connection.
lastResultTime	Time of the last attempt to establish the connection.

Response Codes

200	OK
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /network/ipsec/connections/myconnection/diagnostics


```
{
    "name": "myconnection",
    "tunnelStatus": "established",
    "localIpAddress": "10.27.30.40",
    "localVirtualIpAddress": "10.77.88.99",
    "remoteIpAddress": "16.20.30.40",
    "remoteSubnet": "192.168.1.0/24",
    "establishedTime": "10 seconds",
    "lifetime": "24h",
    "phase1Algorithm": "AES_CBC_256/HMAC_SHA2_512_256/PRF_HMAC_SHA2_512/MODP_2048",
    "phase2Algorithm": "AES_CBC_256/HMAC_SHA2_512_256",
    "certificateName": "",
    "connectionStatus": "Security Associations (1 up, 0 connecting):\nmyconnection[1]: ESTABLISHED 10 seconds ago,
10.27.30.40[10.27.30.40]...16.20.30.40[10.70.10.90]\nmyconnection[1]: IKEv2 SPIs: 1f0c2b4117f92cb5_i* 6cbf2e4537257eff_
r, rekeying in 23 hours\nmyconnection[1]: IKE proposal: AES_CBC_256/HMAC_SHA2_512_256/PRF_HMAC_SHA2_
512/MODP_2048\nmyconnection{1}:  INSTALLED, TUNNEL, reqid 1, ESP in UDP SPIs: c365ad26_i 28e94563_
o\nmyconnection{1}:  AES_CBC_256/HMAC_SHA2_512_256, 0 bytes_i, 0 bytes_o, rekeying in 54 minutes\nmyconnection{1}:
  10.77.88.99/32 === 192.168.1.0/24\n","lastResult": "[IKE] initiating IKE_SA myconnection[1] to 16.20.30.40\n[ENC]
generating IKE_SA_INIT request 0 [ SA KE No N(NATD_S_IP) N(NATD_D_IP) N(FRAG_SUP) N(HASH_ALG) N(REDIR_
SUP) ]\n[NET] sending packet: from 10.27.30.40[500] to 16.20.30.40[500] (1136 bytes)\n[NET] received packet: from
16.20.30.40[500] to 10.27.30.40[500] (416 bytes)\n[ENC] parsed IKE_SA_INIT response 0 [ SA KE No N(NATD_S_IP) N
(NATD_D_IP) ]\n … [IKE] CHILD_SA myconnection{1} established with SPIs c365ad26_i 28e94563_o and TS 10.77.88.99/32
=== 192.168.1.0/24\ninitiate completed successfully\n",
    "lastResultTime": "Sat Dec 03 04:15:15 UTC 2022"
}
```

2.4.13 /network/ipsec/certificates

Use this resource to list and delete IPSec PKCS12 files.

Methods

GET, DELETE

Parameters

None.

Request Body

None.

Response Body

Array of IPSec certificate names:

Parameter	Description
name	Name of a PKCS12 file.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /network/ipsec/certificates
    {
        "ipsecCertificates": [
            {
                "name": "cert1.p12",
            },
            {
                "name": "cert2.p12",
            }
        ]
    }
```

```
DELETE /network/ipsec/certificates/cert1.p12
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.14 /network/ipsec/certificates/download

Use this resource to download an IPSec certificate from a remote server for the appliance.

NOTE: The certificate must be in PKCS12 format.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
protocol	Specifies the protocol to use to download file: ftp / scp / sftp
ipAddress	IP address of the remote server from which to download the file.
username	Username to access the remote server.
password	Password to access the remote server.
directory	Directory path on the remote server, typically relative to the ftp root directory.
filename	Filename of the certificate file on the remote server.
passphrase	(Optional) Pass phrase used to protect the PKCS12 file.

Response Body

Parameter	Description
status	Status result, typically “ipsec certificate download successful” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /network/ipsec/certificates/download
    {
        "protocol": "ftp",
        "ipAddress": "10.20.30.80",
        "username": "anonymous",
        "password": "anonymous",
        "directory": "pub/certificate/",
        "filename": "mycertificate.p12",
        "passphrase": "mypassphrase"
    }
```

Response is:

```
    {
        "status": "ipsec certificate download successful",
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.15 /network/gre/tunnels[/<NAME>]

Use this resource to configure GRE tunnels and settings.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Specifies the name of the GRE tunnel to access.

Request Body

Parameter	Description
name	GRE tunnel name.
tunnelStatus	Read-only current tunnel status: up / down
localIpAddress	IP address of the local interface.
remoteIpAddress	IP address of the remote interface.
localTunnelNetwork	IP address and prefix of the local tunnel network. Example: 10.0.0.1/24
remoteTunnelIpAddress	Optional IP address on the remote tunnel network.
bootAction	Specifies whether this tunnel will be established at boot time: yes / no

Response Body

Same as request body.

Response Codes

200	OK
201	Created
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /network/gre/tunnels
    {
        "greTunnels": [
            {
```

```
                "name": "gre11",
                "tunnelStatus": "up",
                "localIpAddress": “10.20.0.40”,
                "remoteIpAddress": “10.20.0.50",
                "localTunnelNetwork": "10.0.0.1/24",
                "remoteTunnelIpAddress": "10.0.0.2",
                "bootAction": "yes"
            },
            {
                "name": "gre12",
                "tunnelStatus": "down",
                "localIpAddress": “10.30.0.40”,
                "remoteIpAddress": “10.30.0.50",
                "localTunnelNetwork": "10.0.10.1/24",
                "remoteTunnelIpAddress": "10.10.0.2",
                "bootAction": "no"
            }
        }
    }
```

GET /network/gre/tunnels/gre11

```
    {
        "name": "gre11",
        "tunnelStatus": "up",
        "localIpAddress": “10.20.0.40”,
        "remoteIpAddress": “10.20.0.50",
        "localTunnelNetwork": "10.0.0.1/24",
        "remoteTunnelIpAddress": "10.0.0.2",
        "bootAction": "yes"
    }
```

DELETE /network/gre/tunnels/gre11

PUT /network/gre/tunnels/gre11

```
    {
        "remoteIpAddress": "10.20.0.41",
        "bootAction": "no"
    }
```

POST /network/gre/tunnels

```
    {
        "name": "gre33",
        "localIpAddress": “10.20.0.40”,
        "remoteIpAddress": “10.20.0.50",
        "localTunnelNetwork": "10.0.0.1/24",
        "remoteTunnelIpAddress": "10.0.0.2",
        "bootAction": "no"
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

Response is:

```
{
    "name": "gre33",
    "localIpAddress": “10.20.0.40”,
    "remoteIpAddress": “10.20.0.50",
    "localTunnelNetwork": "10.0.0.1/24",
    "remoteTunnelIpAddress": "10.0.0.2",
    "bootAction": "no"
}
```

2.4.16 /network/gre/tunnels/<NAME>/connect

Use this resource to establish the specified GRE tunnel.

Methods

POST

Parameters

Parameter	Description
NAME	Specifies the name of the GRE tunnel to establish.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “initiated connect of <NAME>.” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /network/gre/tunnels/gre11/connect
```

Response is:

```
{
    "status": "initiated connect of gre11."
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.17 /network/gre/tunnels/<NAME>/disconnect

Use this resource to delete the specified GRE tunnel.

Methods

POST

Parameters

Parameter	Description
NAME	Specifies the name of the GRE tunnel to disconnect.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “initiated disconnect of <NAME>.” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /network/gre/tunnels/gre11/disconnect
```

Response is:

```
{
    "status": "initiated disconnect of gre11."
}
```

2.4.18 /network/gre/tunnels/<NAME>/ping

Use this resource to attempt to test a GRE tunnel using a network ping.

Methods

POST

Parameters

Parameter	Description
NAME	Specifies the name of the GRE tunnel connection to access.

Request Body

Parameter	Description
ipAddress	The IP address to attempt to ping on the other side of the GRE tunnel.
count	Number of pings to send between 1 and 10 (integer). Default: 3

Response Body

Parameter	Description
status	Status result, typically “success” on success.
detail	Detailed ping command output.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /network/gre/tunnels/gre11/ping
    {
        "ipAddress": "192.168.1.99",
        "count": 3
    }
```

Response is:

```
    {
        "status": "success",
        "detail": "PING 192.168.7.99 (192.168.7.99) 56(84) bytes of data.\n64 bytes from 192.168.7.99: icmp_seq=1 ttl=255 time
        =112 ms\n\n--- 192.168.7.99 ping statistics ---\n1 packets transmitted, 1 received, 0% packet
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
loss,time0ms\nrttmin/avg/max/mdev = 112.128/112.128/112.128/0.000 ms\n"
    }
```

2.4.19 /network/gre/tunnels/<NAME>/diagnostics

Use this resource to read information about the specified GRE tunnel.

Methods

GET

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Specifies the name of the GRE tunnel to access.

Request Body

None.

Response Body

Parameter	Description
name	GRE tunnel name.
route	Show the route for the GRE tunnel.
tunnel	Show the GRE tunnel attributes.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /network/gre/tunnels/gre11/diagnostics
    {
        "name": "gre11",
        "route": "10.0.0.0/24 dev gre11 proto kernel scope link src 10.0.0.1 ",
        "tunnel": "gre11: gre/ip remote 10.20.0.50 local 10.20.0.40 ttl inherit"
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.20 /network/snmp[/<ID>]

Use this resource to configure SNMP settings.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
ID	Specifies the SNMP ID to access.

Request Body

Parameters	Description
id	Read-only identifier composed of name and potentially source fields.
name	SNMP community name for v1/v2 or the username for v3.
version	SNMP version: **v1_v2** / **ipv6_v1_v2** / **v3**
source	Subnet address for v1/v2.
oid	Unique snmp identifier.
permission	Access permission: **read_and_write** / **read_only**
authenticationType	SNMP v3 authentication type: **md5** / **sha** / **sha256** / **sha512**
authenticationPassphrase	SNMP v3 authentication passphrase.
encryptionMethod	SNMP v3 encryption method: **des** / **aes** / **aes192** / **aes256**
privacyPassphrase	SNMP v3 privacy passphrase.
minimumSecurityLevel	SNMP v3 minimum security level: **auth_no_priv** / **auth_priv** / **no_auth_no_priv**

Response Body

Same as request body.

Response Codes

200	OK
201	Created
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /network/snmp

```
{
    "snmp": [
        {
            "id": "mycommunity~10.20.30.40",
            "name": "mycommunity",
            "version": "v1_v2",
            "source": "10.20.30.40",
            "oid": "",
            "permission": "read_only"
        },
        {
            "id": "myusername",
            "name": "myusername",
            "version": "v3",
            "oid":"",
            "permission": "read_only",
            "authenticationType": "md5",
            "authenticationPassphrase": ****,
            "encryptionMethod": "des",
            "privacyPassphrase": ****,
            "minimumSecurityLevel": "no_auth_no_priv"
        }
        …
    ]
}
```

GET /network/snmp/1

```
{
    "id": "mycommunity~10.20.30.40",
    "name": "mycommunity",
    "version": "v1_v2",
    "source": "10.20.30.40"
    "oid": "",
    "permission": "read_only",
}
```

DELETE /network/snmp/mycommunity~10.20.30.40

GET /network/snmp/myusername

```
{
    "id": "myusername",
    "name": "myusername",
    "version": "v3",
    "oid": "",
    "permission": "read_only",
    "authenticationType": "md5",
    "authenticationPassphrase": ****,
    "encryptionMethod": "des",
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
        "privacyPassphrase": ****,
        "minimumSecurityLevel": "no_auth_no_priv"
    }
```

```
POST /network/snmp
    {
        "name": "mycommunity",
        "oid": "",
        "permission": "read_only",
        "version": "v1_v2",
        "source": "10.20.30.40"
    }
```

Response is:

```
    {
        "id": "mycommunity~10.20.30.40",
        "name": "mycommunity",
        "version": "v1_v2",
        "source": "10.20.30.40"
        "oid": "",
        "permission": "read_only",
    }
```

```
PUT /network/snmp/mycommunity~10.20.30.40
    {
        "oid": ".1",
        "permission": "read_write",
    }
```

2.4.21 /network/snmp/system

Use this resource to configure SNMP system settings.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
sysContact	SNMP system contact information. Default: Avocent_Corporation
sysLocation	SNMP system location information. Default: Avocent_ACS8000

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /network/snmp/system
    {
        "sysContact": "Avocent_Corporation",
        "sysLocation": "Avocent_ACS8000"
    }
```

```
PUT /network/snmp/system
    {
        "sysContact": "corporate IT @ 1-800-xxx-yyyy",
        "sysLocation": "Server room 123"
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.22 /network/dhcpServerSettings

Use this resource to configure the DHCP Server settings.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
status	DHCP server status: **enabled** / **disabled**
interface	The network interface which listens for DHCP requests: **eth**#
subnetAddress	Defines the subnet address which will be used for address assignments.
subnetMask	Defines the subnet mask which will be used for address assignments.
defaultLeaseTime	The time, in hours, which will be assigned to a lease if the client does not request an expiration time.
maxLeaseTime	The maximum time, in hours, which will be assigned to a lease.
startIpAddress	Start of the range of IP addresses used for random IP address assignment.
endIpAddress	End of the range of IP addresses used for random IP address assignment.
vendorClassId	(Optional) If a vendor class identifier is specified, then only devices which report this identified will be assigned an address from the address pool.
serverName	(Optional) Used for Zero Touch Provisioning.
filename	(Optional) Used for Zero Touch Provisioning.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /network/dhcpServerSettings

```
{
    "status": "enabled",
    "interface": "eth1",
    "subnetAddress": "192.168.161.0",
    "subnetMask": "255.255.255.0",
    "defaultLeaseTime": "12",
    "maxLeaseTime": "12",
    "startIpAddress": "192.168.161.100",
    "endIpAddress": "192.168.161.200",
    "vendorClassId": "",
    "serverName": "",
    "fileName": ""
}
```

PUT /network/dhcpServerSettings

```
{
    "maxLeaseTime": "24",
    "startIpAddress": "192.168.161.10",
    "endIpAddress": "192.168.161.50"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.23 /network/dhcpServerAddresses[/<IPADDRESS>]

Use this resource to return information about all DHCP assigned addresses, both dynamically assigned addresses and reserved addresses, as well as to add new reserved addresses and modify existing reserved addresses.

NOTE: Dynamically assigned addresses may not be modified or added.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields and filtering queries are supported for all body parameters.

Parameter	Description
IPADDRESS	Specifies the IP address to access.

Request Body

Parameter	Description
ipAddress	IP address assigned to the host by the DHCP server. **NOTE: Writeable only for reserved addresses.**
macAddress	MAC address of the host. **NOTE: Writeable only on a POST to the resource when the reserved address entry is created.**
hostName	Name of the host. **NOTE: Writeable only for reserved addresses.**
vendorId	Read-only. Vendor ID reported by the host.
serverName	(Optional field for reserved address assignments) Used for Zero Touch Provisioning. **NOTE: Writeable only for reserved addresses**
fileName	(Optional field for reserved address assignments) Used for Zero Touch Provisioning. **NOTE: Writeable only for reserved addresses.**
reserved	Read-only. Set to **yes** if this is a reserved address; otherwise, set to **no**.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /network/dhcpServerAddresses
    {
        "addresses": [
            {
                "ipAddress": "192.168.161.100",
                "macAddress": "ee:e0:86:14:54:a6",
                "hostName": "RPC2-551831G100C2014DEC180012",
                "vendorId": "udhcp 1.5.0",
                "reserved": "no"
            },
            {
                "ipAddress": "192.168.161.101",
                "macAddress": "ee:e0:86:11:25:14",
                "hostName": "apc112514",
                "vendorId": "APC",
                "reserved": "no"
            },
            {
                "ipAddress": "192.168.161.45",
                "macAddress": "ee:e0:86:23:29:33",
                "hostName": "test",
                "serverName": "192.168.1.1",
                "fileName": "ztp_tftp.py",
                "reserved": "yes"
            }
        ]
    }
```

```
PUT /network/dhcpServerAddresses/192.168.161.45
    {
        "hostName": "test",
        "serverName": "newserver",
        "fileName": "ztp_tftp1.py"
    }
```

To add a reserved DHCP Address:

```
POST /network/dhcpServerAddresses
    {
        "ipAddress": "192.168.161.45",
        "macAddress": "ee:e0:86:23:29:33",
        "hostName": "test"
    }
```

Response is:

```
    {
        "ipAddress": "192.168.161.45",
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
        "macAddress": "ee:e0:86:23:29:33",
        "hostName": "test"
    }
```

To remove a reserved DHCP address:

```
DELETE /network/dhcpServerAddresses/192.168.161.45
```

2.4.24 /network/certificates/settings

Use this resource to return information about the certificate management settings.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
autoUpdate	Control whether certificate management automation is enabled: **disabled** / **acmev2**
acmeSettings	
domain	Domain name being requested for the certificate.
directory	ACME server endpoint for client connections.
emailAddress	Contact address sent to the ACME server.
challengeType	Specifies the challenge type used for domain name validation: **http-01** / **tls-alpn-01** / **dns-01**

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /network/certificates/settings

```
{
    "autoUpdate": "disabled",
    "acmeSettings": {
        "domain": "test.com",
        "directory": "http://test.com",
        "emailAddress": "",
        "challengeType": "dns-01"
    }
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
PUT /network/certificates/settings
        {
            "autoUpdate": "acmev2",
            “acmeSettings”: {
                “domain”: “test.com”,
                “challengeType”:”dns-01”
            }
        }
```

2.4.25 /network/certificates/status

Use this resource to return status information on certificate management.

Methods

GET

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
domain	Current certificate's domain name.
status	Current certificate's status, typically "good" if the certificate is functioning normally.
validFrom	Date and time that the current certificate is valid from.
validTo	Date and time that the current certificate is valid to.
renewal	Date and time when the next renewal is scheduled.
output	Current state of the ACME client and the downloaded certificate.
logFile	Log messages passed between the ACME client and the server during the most recent renewal process.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /network/certificates/status
        {
            "domain": "acs804.acsdemo.acs",
            "status": "good",
            "validFrom": "2023-01-05 19:46:03",
            "validTo": "2023-01-06 19:47:03",
            "renewal": "2023-01-06 11:51:32",
            "output": "{\n  \"name\": \"acs804.acsdemo.acs\",\n  \"domains\": [\n    \" acs804.acsdemo.acs \"\n  ],\n  \"contacts\": [\n
            \"mailto:you@example.com\"\n  ],\n  \"transitive\": 1,\n  \"ca\": {\n    \"account\": \"ACME-10.20.30.40-0000\",\n
            \"proto\": \"ACME\",\n    \"url\": \"https://10.20.30.40/acme/acme/directory\",\n    \"urls\": [\n
            \"https://10.20.30.40/acme/acme/directory\"\n    ],\n    \"agreement\": \"accepted\"\n  },\n  \"privkey\": {\n    \"type\":
            \"RSA\",\n    \"bits\": 2048\n  },\n  \"state\": 2,\n  \"renew-mode\": 2,\n  \"renew-window\": \"33%\",\n  \"warn-window\":
```

 Proprietary and Confidential ©2025 Vertiv Group Corp.

\"10%\",\n \"must-staple\": false,\n \"proto\": {\n   \"acme-tls/1\": [\n   \" acs804.acsdemo.acs \"\n   ]\n },\n \"stapling\": false,\n \"cert\": {\n   \"rsa\": {\n   \"valid\": {\n   \"from\": \"Thu, 14 Nov 2024 16:41:29 GMT\",\n \"until\": \"Fri, 15 Nov 2024 16:42:29 GMT\"\n   },\n   \"serial\": \"6AA…179\",\n   \"sha256-fingerprint\": \"669ad29d…0fc90f129\"\n   },\n   \"valid\": {\n   \"from\": \"Thu, 14 Nov 2024 16:41:29 GMT\",\n   \"until\": \"Fri, 15 Nov 2024 16:42:29 GMT\"\n   }\n },\n \"renew-at\": \"Fri, 15 Nov 2024 08:46:58 GMT\",\n \"watched\": true\n}",",
"logFile": "{\n \"name\": \" acs804.acsdemo.acs \",\n \"domains\": [\n   \" acs804.acsdemo.acs \"\n],\n \"contacts\": [\n   \"mailto:you@example.com\"\n],\n \"transitive\": 1,\n \"ca\": {\n   \"account\": \"ACME-10.207.24.151-0000\",\n \"proto\": \"ACME\",\n   \"url\": \"https://10.207.24.151/acme/acme/directory\",\n   \"urls\": [\n \"https://10.20.30.40/acme/acme/directory\"\n   ],\
...
      \"when\": \"Thu, 14 Nov 2024 16:41:58 GMT\",\n   \"type\": \"progress\",\n   \"detail\": \"Assessing current status\"\n   },\n   {\n   \"when\": \"Thu, 14 Nov 2024 16:41:58 GMT\",\n   \"type\": \"progress\",\n   \"detail\": \"Resetting staging area\"\n   },\n   {\n   \"when\": \"Thu, 14 Nov 2024 16:41:58 GMT\",\n   \"type\": \"progress\",\n   \"detail\": \"Checking staging area\"\n   },\n   {\n   \"when\": \"Thu, 14 Nov 2024 16:41:58 GMT\",\n   \"type\": \"progress\",\n   \"detail\": \"MDomain acs804.acsdemo.acs initialized with support for ACME challenges tls-alpn-01 http-01 dns-01\"\n   },\n   {\n   \"when\": \"Thu, 14 Nov 2024 16:41:58 GMT\",\n \"type\": \"starting\"\n   },\n   {\n   \"when\": \"Thu, 14 Nov 2024 16:41:58 GMT\",\n   \"type\": \"message-renewing\"\n   }\n   ]\n }\n}",

2.4.26 /network/certificates/root

Use this resource to list network root certificate files.

Methods

GET

Parameters

None.

Request Body

None.

Response Body

Array of root certificate names:

Parameter	Description
name	Name of a certificate file.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /network/certificates/root
    {
        "networkCertificates": [
            {
                "name": "root1.crt",
            },
            {
                "name": "myrootCA.crt",
            }
        ]
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.27 /network/certificates/root/<NAME>

Use this resource to get the details of a specific root certificate or to delete a specific root certificate.

Methods

GET, DELETE

Parameters

Format query is supported and applies to the data parameter. Formats include text and base64.

Request Body

None.

Response Body

Parameter	Description
data	Data for the certificate. Certificate information is returned as a text summary of the certificate information.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /network/certificates/root/mycert.crt
    {
        "data": " "Filename: mycert.crt\n\nCertificate:\n    Data:\n        Version: 3 (0x2)\n
        Serial Number:\n        63:e6:5a:7d:e0:8e:70:b3:44:dd:c8:bf:de:d9:02:7a\n        Signature Algorithm:
        sha256WithRSAEncryption\n        Issuer:CN = DCIMAccess-CAAUTH-DCIM-CA-Root\n        Validity\n
        Not Before: Apr 23 19:15:05 2019 GMT\n            Not After : Apr 23 19:25:04 2027 GMT\n        Subject: CN = DCIMAccess
        -CAAUTH-DCIM-CA-Root\n        Subject Public Key
        …
        95:40:58:9f:58:41:56:e5:c5:46:99:69:16:38:8b:03:07:f1:\n        d9:b5:6e:0d:b8:53:35:1e:7e:62:4a:3f:0c:be:48:8b:f4:05:\n
        55:83:9f:cd\n"
    }

GET /network/certificates/root/mycert.crt?format=base64
    {
        "data":
        "RmlsZW5hbWU6IGJpbGwuY3J0CgpDZXJ0aWZpY2F0ZToKICAgIERhdGE6CiAgICAgICAgVmVyc2lvbjogMyAoMHg
        yKQogICAgICAgIFNlcmlhbCBOdW1iZXI6CiAgICAgICAgICAgIDYzOmU2OjVhOjdkOmUwOjhlOjcwOmIzOjQ0OmRkO
        mM4
        …
        Q5OmI1OjZlOjBkOmI4OjUzOjM1OjFlOjdlOjYyOjRhOjNmOjBjOmJlOjQ4OjhiOmY0OjA1OgogICAgICAgIDU1OjgzOjlm
```

```
    OmNkCg=="
}
```

```
DELETE /network/certificates/root/myrootCA.crt
```

2.4.28 /network/certificates/root/download

Use this resource to download a root certificate from a remote server to the appliance.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
protocol	Specifies the protocol to use to download file: ftp / scp / sftp / https
ipAddress	IP address of the remote server from which to download the file.
username	Username to access the remote server.
password	Password to access the remote server.
directory	Directory path on the remote server, typically relative to the ftp root directory.
filename	Filename of the certificate file on the remote server.

Response Body

Parameter	Description
status	Status result, typically “network certificate download successful” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /network/certificates/root/download
    {
        "protocol": "ftp",
        "ipAddress": "10.20.30.80",
        "username": "anonymous",
        "password": "anonymous",
        "directory": "pub/certificate/",
        "filename": "myrootcert.pem",
    }
```

Response is:

```
{
    "status": "network certificate download successful",
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.4.29 /network/certificates/root/upload

Use this resource to upload a root certificate file contained in the body of the JSON message.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
filename	Filename used when saving this certificate on the console system.
data	Base64 encoded data for the certificate.

Response Body

Parameter	Description
status	Status result, typically “network certificate upload successful” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /network/certificates/root/upload
    {
        “filename”: “myrootcert.pem”,
        "data":  "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURMVENDQWhXZ0
            ...
            scEZqaUxBd2Z4MmJWdURiaFROUjUrWWtvL0RMNUlpL1FGVlIPZgp6UT09Ci0t
            LS0tRU5EIENFUlRJRklDQVRFLS0tLS0K"
    }
```

Response is:

```
    {
        "status": "network certificate upload successful",

    }
```

2.5 Serial Ports

2.5.1 /serialPorts[/<PORT>]

Use this resource to get and set various serial port parameters for one or more serial ports.

Methods

GET, PUT, PATCH

Parameters

Fields and filtering queries are supported for all body parameters.

Parameter	Description
PORT	The number of the serial port to access (integer).

Request Body

Parameter	Description
port	Read-only port number.
profile	Port profile: **cas** / **power** / **dial_in** / **dial_out** / **socket_client** / **unconfigured** **NOTE: Unconfigured is a read-only state that cannot be set.**
deviceName	Read-only device name assigned by the operating system. Examples: ttyS1, ttyUSB0, ttyACM0.
status	Port status: **enabled** / **disabled**
target	Read-only target status: **on** / **off** / **unknown**
physical (only valid when the profile is set to "cas", "power" or "socket_client")	
pinout	Serial port pinout: **auto** / **cisco** / **cyclades** / **rs422** / **rs485**
speed	Serial port speed (integer): **300** / **600** / **1200** / **1800** / **2400** / **4800** / **9600** / **19200** / **38400** / **57600** / **115200** / **230400**
parity	Parity: **even** / **odd** / **none**
dataBits	Number of data bits (integer): **5** / **6** / **7** / **8**
stopBits	Number of stop bits (integer): **1** / **2**
flowControl	Flow control mechanism: **none** / **hardware** / **software** / **rxon_software** / **txon_software**
cas (only valid when the profile is set to "cas")	
name	User-assigned or default name for the port. Default is the last half of the appliance MAC address followed by "-p-" and the port number. Example: 10-2c-3d-p-1
autoDiscovery	Enables auto discovery: **enabled** / **disabled** The target name is discovered and associated with this serial port. If it fails, the default port name is used.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
autoDiscoveryType	Type of auto discovery to employ: **probe** / **command** Default: traditional probe style. When configured to "command" the appliance attempts to login to the target device using the autoDiscovery username and password values provided and executes the command specified to try and determine the hostname of the target device.
autoDiscoveryUsername	Username to access the target device.
autoDiscoveryPassword	Password to access the target device.
autoDiscoveryCommand	Command to use to determine the hostname of this target device. This is one of the command names defined in the /casProfile/commands table.
speedAutoDetection	Enables speed auto detection to try to discover the speed of the serial port: **enabled** / **disabled**
protocol	The protocol used by authorized users to access the serial port or target: **ssh** / **telnet** / **raw_mode** / **telnet|ssh** / **telnet|raw_mode** / **ssh|raw_mode** / **telnet|ssh|raw_mode**
authentication	Type used to authenticate a user during the target session: **none** / **dsview_down_local** / **dsview** / **dsview|local** / **kerberos_down_local** / **kerberos** / **kerberos|local** / **ldap_down_local** / **ldap** / **ldap|local** / **local** / **local|ldap** / **local|radius** / **localtacacs+** / **otp** / **otp|local** / **radius_down_local** / **radius** / **radius|local** / **tacacs+_down_local** / **tacacs+** / **tacacs+|local**
textHotKey	Hotkey to suspend the target session and return to the cli prompt. Not available for raw. Default: Ctrl-Z (^Z)
powerHotKey	Hotkey to suspend the target session and display the Power Management Menu to control the outlets merged to the target. Not available for raw. Default: Ctrl-P (^P)
restfulHotKey	Hotkey to suspend the target session and display the RESTful Client Menu, which is used to send user-defined RESTful actions to a RESTful server. Default: blank.
telnetAliasPort	TCP port used to connect directly to a serial port using Telnet protocol. Default: 7000+ port number
sshAliasPort	TCP port used to connect directly to a serial port using SSH protocol. Default: blank.
rawModeAliasPort	TCP port used to connect directly to a serial port using raw socket for connection. Default: blank.
ipv4AliasAddress	IPv4 address used to connect directly to a serial port.
ipv4AliasInterface	Interface associated with the ipv4AliasAddress: **eth#**
ipv6AliasAddress	IPv6 address used to connect directly to a serial port.
ipv6AliasInterface	Interface associated with the ipv6AliasAddress: **eth#**
dcdSensitivity	Allows a session only if DCD is on: **enabled** / **disabled**
autoAnswer	Enables processing of input data so when the input data matches one input string configured in Auto Answer, the configured output string is transmitted to the serial port: **enabled** / **disabled**
dtrMode	Sets DTR mode. **always_on** – DTR status is always on. **normal** (default) – DTR status depends on the existence of a CAS session. **off_interval** - when the CAS session is closed. DTR stays down during this interval.
dtrOffInterval	Interval used by DTR Mode off_interval in milliseconds (integer). Default: 100

Parameter	Description
linefeedSuppression	Enables the suppression of the line-feed character after the carriage return character: **enabled**/**disabled**
nullAfterCrSuppression	Enables the suppression of the NULL character after the carriage return character: **enabled**/**disabled**
transmissionInterval	The interval the port waits to send data to a remote client in milliseconds (integer).
breakSequence	An administrator can configure the control key as the break sequence, entering ^ before the letter. Default: ~break
breakInterval	Interval for the break signal in milliseconds (integer). Default: 500
multiSessionMenu	Enables the multi-session menu when connecting to a port that is already being accessed by another user: **enabled** / **disabled**
loginNotification	Enables the notification to multi-session users when a user logs in or logs out: **enabled** / **disabled**
infoNotification	Displays an information message when a target session is opened: **enabled** / **disabled**
alerts	Enables detection of alerts. See /serialPorts/<PORT>/alerts to configure alerts: **enabled** / **disabled**
cas/dataBuffering (only valid when the profile is set to "cas")	
bufferingStatus	Enables or disables data buffering: **enabled** / **disabled**
bufferingType	Controls the type of data buffering: **local** – stores the data buffering file on the appliance's local file system **nfs** – stores the data buffering file on an NFS server **syslog** – sends the data to the syslog server **dsview** – sends the data to the Vertiv™ management software.
localDevice	When the bufferingType is set to local, this field specifies where on the local system the data buffering files are stored: the built-in memory (**mmcblk0**) or a connected USB storage or SD card location.
timeStamp	When enabled, adds the time stamp to the data buffering line for a local or NFS server: **enabled** / **disabled**
loginMessage	Includes special notification for logins and logouts in data buffering: **enabled** / **disabled**
sessionLogging	Controls when data is stored: **enabled** / **disabled**
power (only valid when the profile is set to "power")	
speedAutoDetect	Enables speed auto detect for power device. Tries to discover the speed of the attached power device: **enabled** / **disabled**
pollingRate	Interval in seconds to update information from the PDU (integer). Default: 20
deviceType	Type of power device connected to the serial port: **pdu** / **ups**
pduType	Defines the type or vendor of the PDU connected to the serial port: **auto** / **cyclades** / **enp** / **spc** / **servertech** / **raritan** / **apc** / **eaton** / **geist**
powerCycleInterval	The interval in seconds between Off and On actions for the power cycle command (integer). Default: 15
syslog	When enabled, the PDU sends syslog messages to the appliance: **enabled** / **disabled**
buzzer	Enables or disables the PDU's buzzer: **enabled** / **disabled**
overcurrentProtection	When enabled, the software's overcurrent protection is on: **enabled** / **disabled**

Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
upsType	Defines the type or vendor of the UPS connected to the serial port: auto / gxt4 / gxt5
socketClient (only valid when the profile is set to "socket_client")	
remoteServer	IPv4 or IPv6 address of the remote server.
remoteTcpPort	TCP port to be used to establish a connection with a remote server.
establishConnection	Configures the event that triggers the establishment of the connection: dcd / always

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /serialPorts/1

```
{
    "port": "1",
    "profile": "cas",
    "deviceName": "ttyS1",
    "status": "enabled",
    "target": "on",
    "physical": {
        "pinout": "cisco",
        "speed": 115200,
        "parity": "none",
        "dataBits": 8,
        "stopBits": 1,
        "flowControl": "none"
    },
    "cas": {
        "name": "MyServer",
        "autoDiscovery": "disabled",
        …
    }
}
```

GET /serialPorts

```
{
    "serialPorts": [
        {
            "port": "1",
            "profile": "cas",
            "deviceName": "ttyS1",
            "status": "enabled",
            …
        },
        …
        {
            "port": "48",
            …
        }
    ]
}
```

PUT /serialPorts/1 '{"cas":{"sshAliasPort":"8001"}}'

GET /serialPorts/75

```
{
    "error": {
        "code": "AE002",
        "message": "resource id not found",
        "detail": "75 is not a valid port id"
    }
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.5.2 /serialPorts/<PORT>/enable

Use this resource to enable the serial port.

Methods

POST

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “success. port <PORT> enabled.” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /serialPorts/3/enable
```

Response is:

```
{
    "status": "success. port 3 enabled."
}
```

2.5.3 /serialPorts/<PORT>/disable

Use this resource to disable the serial port.

Methods

POST

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “success. port <PORT> disabled.” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /serialPorts/3/disable
```

Response is:

```
{
    "status": "success. port 3 disabled."
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.5.4 /serialPorts/<PORT>/clone

Use this resource to clone the settings of the source port specified in the URL to a list of one or more other port numbers.

Methods

POST

Parameters

Parameter	Description
PORT	The integer number of the serial port to be cloned.

Request Body

Parameter	Description
ports	List of port numbers to which to copy the original port settings. The list can contain comma separated port numbers and ranges of port numbers with two numbers separated using a hyphen (-). Example: 2,4-6,8

Response Body

Parameter	Description
status	Status result, typically “success. port <PORT> cloned to port(s) <PORTS>.” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /serialPorts/1/clone
    {
        "ports": "2,4-6,8"
    }
```

Response is:

```
    {
        "status": "success. port 3 cloned to port(s) 2,4-6,8.",
    }
```

2.5.5 /serialPorts/<PORT>/resetToFactory

Use this resource to reset the serial port to factory defaults.

Methods

POST

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “success. port <PORT> restored to factory." on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /serialPorts/3/resetToFactory
```

Response is:

```
{
    "status": "success. port 3 restored to factory."
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.5.6 /serialPorts/<PORT>/discoverNow

Use this resource to initiate the Auto Discovery procedure to determine the port name for a serial port.

Methods

POST

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “success. port <PORT> auto discovery initiated." on success.

Response Codes

200	OK
400	Bad Request

Examples

POST /serialPorts/3/discoverNow

Response is:

```
{
    "status": "success. port 3 auto discovery initiated."
}
```

2.5.7 /serialPorts/<PORT>/alerts

Use this resource to configure alert strings.

Methods

GET, POST

Parameters

Fields and filtering queries are supported for all body parameters.

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

Parameter	Description
index	Read-only index (integer).
alertString	String to match.
scriptName	Name of the script to run when the matching alertString is detected. This script should be placed in the /etc/alert_scripts folder. The scriptName should be the filename only, with a '.sh' extension and no folder path.
emergency	Signal using the amber led on the serial port: **yes/no** If set to 'yes' and an alert occurs, the serial port LED blinks amber quickly.

Response Body

Same as request body.

Response Codes

200	OK
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
GET /serialPorts/3/alerts
```

Response is:

```
{
    "alertStrings": [
    {
        "index": 1,
        "alertString": "Invalid",
        "scriptName": "sample_script.sh",
        "emergency": "no"
    },
    {
        "index": 2,
        "alertString": "Error occurred",
        "scriptName": "sample_script.sh",
        "emergency": "yes"
    },
    ]
}
```

To add a new alert string:

```
POST /serialPorts/3/alerts
    {
        "alertString": "new string",
        "scriptName": "new_script.sh",
        "emergency": "yes"
    }
```

Response is:

```
{
    "index": 1,
    "alertString": "new string",
    "scriptName": "new_script.sh",
    "emergency": "yes"
}
```

2.5.8 /serialPorts/<PORT>/alerts/<ID>

Use this resource to delete an alert string.

Methods

DELETE

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.
ID	The integer index of the alert string to access.

Request Body

None.

Response Body

None.

Response Codes

200	OK
400	Bad Request

Examples

```
DELETE /serialPorts/3/alerts/1
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.5.9 /serialPorts/<PORT>/alerts/clear

Use this resource to clear active alerts on a serial port.

Methods

POST

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “success. alerts cleared.” on success.

Response Codes

200	OK
400	Bad Request

Examples

POST /serialPorts/3/alerts/clear

Response:

```
{
    "status": "success. alerts cleared."
}
```

2.5.10 /serialPorts/<PORT>/alerts/deleteAny

Use this resource to delete all alert strings for the port.

Methods

POST

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “success. alert strings deleted.” on success.

Response Codes

200	OK
400	Bad Request

Examples

POST /serialPorts/3/alerts/deleteAny

Response:

```
{
    "status": "success. alert strings deleted."
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.5.11 /serialPorts/<PORT>/power

Use this resource to merge PDU outlets or UPS outlet groups with the serial port and view all merged outlets.

NOTE: Either PDU outlets or UPS outlet groups can be merged with a serial port but not a mix of both.

Methods

GET, POST

Parameters

Fields query is supported for all body parameters.

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

Parameter	Description
Id	Read-only ID for the outlet consisting of the PDU name and the outlet number, or the UPS name and the UPS outlet group number, in this format: devicename~outletnumber
deviceType	Identifies the power device: **pdu** or **ups**
deviceId	Name of the PDU or UPS.
outlet	PDU outlet number or UPS outlet group number.

Response Body

Same as request body.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /serialPorts/1/power
    {
        "power": [
            {
                "id": "MPX~3",
                "deviceType": "pdu",
                "deviceId": "MPX",
                "outlet": "3"
            },
            {
                "id": "MPX~4",
                "deviceType": "pdu",
                "deviceId": "MPX",
                "outlet": "4"
            }
        ]
    }
```

```
GET /serialPorts/2/power
    {
        "power": [
            {
                "id": "GXT4~1",
                "deviceType": "ups",
                "deviceId": "GXT4",
                "outlet": "1"
            },
            {
                "id": "GXT4~2",
                "deviceType": "ups",
                "deviceId": "GXT4",
                "outlet": "2"
            }
        ]
    }
```

To merge a new outlet with the serial port:

```
POST /serialPorts/1/power
    {
        "deviceType": "pdu",
        "deviceId": "MPX",
        "outlet": "5"
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

Response is:

```
{
    "id": "MPX~5",
    "deviceType": "pdu",
    "deviceId": "MPX",
    "outlet": "5"
}
```

```
POST /serialPorts/2/power
{
    "deviceType": "ups",
    "deviceId": "GXT4-1000",
    "outlet": "1"
}
```

Response is:

```
{
    "id": "GXT4-1000~1",
    "deviceType": "ups",
    "deviceId": "GXT4-1000",
    "outlet": "1"
}
```

2.5.12 /serialPorts/<PORT>/power/<ID>

Use this resource to delete merged PDU outlets or UPS outlet groups.

Methods

DELETE

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.
ID	The ID of the outlet to access in the following format: devicename~outletnumber

Request Body

None.

Response Body

None.

Response Codes

200	OK
201	Created
400	Bad Request

Examples

```
DELETE /serialPorts/1/power/MPX~5
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.6 Auxiliary Ports

2.6.1 /auxPorts[/<NAME>]

Use this resource to get and set auxiliary port parameters. When a built-in modem is present, there is a single auxiliary port.

NOTE: This resource has been deprecated. The same functionality is now available via the more advanced /modems resource.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of the modem, such as ttyM1.

Request Body

Parameter	Description
deviceName	Example: ttyM1
status	enabled / disabled
profile	unconfigured / dial_in / dial_out
phoneNumber	Dial-Out only.
speed	AuxPort speed (integer): 1200 / 2400 / 4800 / 9600 / 19200 / 38400 / 57600 / 115200
initChat	Dial-In Default: "" \d\d\d+++\d\d\dATZ OK AT+VCID=1 OK
	Dial-Out Default: ABORT BUSY ABORT VOICE ABORT "NO CARRIER" ABORT "NO DIAL TONE: ABORT "NO DIAL TONE" " " ATZ OK ATDT\T CONNECT "" ""
pppAddressConfig	none / local / remote **NOTE: Dial-Out profile only supports local.**
localIpv4Address	IPv4 address of the local side of the ppp connection when pppAddressConfig is local.
remoteIpv4Address	IPv4 address of the remote side of the ppp connection when pppAddressConfig is local.
localIpv6Address	IPv6 address of the local side of the ppp connection when pppAddressConfig is local.

Parameter	Description
remoteIpv6Address	IPv6 address of the remote side of the ppp connection when pppAddressConfig is local.
pppAuthentication	none / appliance / remote
pppAuthenticationProtocol	pap / chap / eap **NOTE: eap is not supported for remote authentication**
pppRemoteUsername	The username for remote authentication.
pppRemotePassphrase	The passphrase for remote authentication.
chapInterval	Number of seconds before rechallenging the peer (integer). Default: 0, disabled.
chapMaxChallenge	Maximum number of challenge transmissions (integer). Default: 10
chapRestart	Restart interval in number of seconds (integer). Default: 3
pppIdleTimeout	Number of seconds (integer). Default: 0

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
GET /auxPorts
        {
            "auxPorts": [
                {
                    "deviceName": "ttyM1",
                    "status": "disabled",
                    "profile": "unconfigured"
                }
            ]
        }
```

```
GET /auxPorts
        {
            "deviceName": "ttyM1",
            "status": "disabled",
            "profile": "dial_in",
            "speed": 38400,
            "initChat": "\"\" \d\d\d+++\d\d\dATZ OK",
            "pppAddressConfig": "none",
            "pppAuthentication": "none",
            "chapInterval": 0,
            "chapMaxChallenge": 10,
            "chapRestart": 3,
            "pppIdleTimeout": 0
        }
```

```
PUT /auxPorts/ttyM1
        {
            "status": "enabled",
            "profile": "dial_out",
            "phoneNumber": "18001234567",
            "speed": 38400,
            "initChat": "ABORT\tBUSY\nABORT\tVOICE\n",
            "localIpv4Address": "10.20.30.40",
            "remoteIpv4Address": "10.20.30.41",
            "localIpv6Address": "",
            "remoteIpv6Address": "",
            "pppAuthentication": "remote",
            "pppAuthenticationProtocol": "pap",
            "pppRemoteUsername": "testuser",
            "pppRemotePassphrase": "mypassphrase",
            "chapInterval": 0,
            "chapMaxChallenge": 10,
            "chapRestart": 3,
            "pppIdleTimeout": 0
        }
```

2.7 Modems

2.7.1 /modems[/<NAME>]

Use this resource to get and set modem parameters. This provides a single resource to configure modem profiles of dial-in and dial-out for all such devices: auxiliary port built-in modem, pluggable USB modems and serial port modems.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of the modem, such as ttyM1.

Request Body

Parameter	Description
deviceName	Ex: ttyM1
status	For analog modem: **disabled** / **enabled** For cellular modem: **disabled** / **failover** / **alwayson**
profile	For analog modem: **unconfigured** / **dial_in** / **dial_out** For cellular modem: **dial_out**
phoneNumber	Dial-Out only.
speed	Modem port speed (integer): **1200** / **2400** / **4800** / **9600** / **19200** / **38400** / **57600** / **115200** Additional speeds for integrated cell modem (integer): **230400** / **460800** / **921600**
initChat	Dial-In Default: "" \d\d\d+++\d\d\dATZ OK AT+VCID=1 OK
	Dial-Out Default: ABORT BUSY ABORT VOICE ABORT "NO CARRIER" ABORT "NO DIAL TONE: ABORT "NO DIAL TONE" " " ATZ OK ATDT\T CONNECT "" ""
pppAddressConfig	**none** / **local** / **remote** (not applicable to cellular) **NOTE: Dial-Out profile only supports local.**
localIpv4Address	IPv4 address of the local side of the ppp connection when pppAddressConfig is local.
remoteIpv4Address	IPv4 address of the remote side of the ppp connection when pppAddressConfig is local.
localIpv6Address	IPv6 address of the local side of the ppp connection when pppAddressConfig is local.

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
remoteIpv6Address	IPv6 address of the remote side of the ppp connection when pppAddressConfig is local.
pppAuthentication	none / appliance / remote
pppAuthenticationProtocol	pap / chap / eap **NOTE: The 'eap' option is not supported for remote authentication.**
pppRemoteUsername	The username for remote authentication.
pppRemotePassphrase	The passphrase for remote authentication.
chapInterval	Number of seconds before rechallenging the peer (integer). Default: 0, disabled
chapMaxChallenge	Maximum number of challenge transmissions (integer). Default: 10
chapRestart	Restart interval in number of seconds (integer). Default: 3
pppIdleTimeout	Number of seconds (integer). Default: 0
INTEGRATED CELLULAR MODEM **NOTE: The following parameters are only present with the integrated cellular modem.**	
model	Vendor model name.
cellProvider	Cellular service provider information from the SIM.
ccid	SIM integrated circuit card identifier.
imsi	International mobile subscriber identity.
imei	International mobile equipment identity.
cellStatus	Status of the integrated cellular modem. Shows if the cell modem is up and connected to the network.
apn	Access point name.
mtuSize	Maximum transmission unit size.
replaceDefaultRoute	Modify the default routes in the routing table: yes / no If this is set to 'yes' then the integrated cellular modem becomes the default gateway.
persistMode	Attempt to bring the modem connection back up if it drops unexpectedly: enabled/disabled
simStatus	Status of the SIM card in the integrated cellular modem: ready / waiting_for_pin / waiting_for_unlock_code / error / unknown
simPin	Enter a new PIN for the SIM card in the integrated cellular modem. Writeable only if simStatus reads *waiting_for_pin*
simUnlockCode	Enter a new unlock code for the SIM card in the integrated cellular modem. Writeable only if simStatus reads *waiting_for_unlock_code*.
epsMode	Sets the EPS mode of operation: default / 0 / 1 / 2 / 3
pdpType	Sets the type of packet data protocol to use in the cellular PDP contexts: default / ip / ipv6 / ipv4v6 / ppp
debugLevel	Sets the debug level to use for cellular modem initialization: 0 / 1 / 2 Default: no additional debugging information, which is level 0. Levels 1 and 2 enable more detail in pppd

Parameter	Description
	logs.
keepAlive	Periodically sends out traffic (a single ping) to keep the connection alive: **enabled** / **disabled**
keepAliveInterval	Interval in seconds for sending out keep alive packet (integer): **10** / **15** / **30** / **60** / **120** / **300** / **600**
keepAliveIpAddress	IP address to which keep alive pings are sent.
useCellProviderDns	Use the DNS entries supplied by the cellular provider instead of the static DNS entries defined in the network settings: **enabled** / **disabled**
inSessionMonitoring	Run the cellular modem in the mode that permits communicating with it for status while it is actively in session: **enabled** / **disabled** Default: disabled (to improve throughput)

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /modems

```
{
    "modems": [
        {
            "deviceName": "ttyM1",
            "status": "disabled",
            "profile": "unconfigured"
        },
        {
            "deviceName": "ttyACM0",
            "status": "disabled",
            "profile": "dial_in",
            …
        },
        {
            "deviceName": "ttyS18",
            "status": "disabled",
            "profile": "dial_out",
            …
        }
    ]
}
```

GET /modems/ttyM1

```
{
    "deviceName": "ttyM1",
    "status": "disabled",
    "profile": "dial_in",
    "speed": 38400,
    "initChat": "\"\" \d\d\d+++\d\d\dATZ OK AT+VCID=1 OK",
    "pppAddressConfig": "none",
    "pppAuthentication": "none",
    "chapInterval": 0,
    "chapMaxChallenge": 10,
    "chapRestart": 3,
    "pppIdleTimeout": 0
}
```

GET /modems/ttyM1

```
{
    "deviceName": "ttyM1",
    "status": "disabled",
    "profile": "dial_out",
    "phoneNumber": "",
    "speed": 921600,
    "initChat": "# Connection script for Wireless GSM/UMTS modems\n…",
    "localIpv4Address": "",
    "remoteIpv4Address": "",
    "localIpv6Address": "",
    "remoteIpv6Address": "",
    "pppAuthentication": "none",
```

```
        "chapInterval": 0,
        "chapMaxChallenge": 10,
        "chapRestart": 3,
        "pppIdleTimeout": 0,
        "model": "Telit LE910C4-NF M0F.660006",
        "cellProvider": "AT\u0026T Wireless Inc., AT\u0026T, USA",
        "ccid": "89012345678912345678",
        "imsi": "310123456789123",
        "imei": "354123456789123",
        "cellStatus": "Disabled, Transceiver off",
        "apn": "m2m.com.attz",
        "mtuSize": "1400",
        "replaceDefaultRoute": "disabled",
        "persistMode": "disabled",
        "simStatus": "ready",
        "simPin": "",
        "simUnlockCode": "",
        "epsMode": "default",
        "pdpType": "default",
        "debugLevel": "0",
        "keepAlive": "enabled",
        "keepAliveInterval": 60,
        "keepAliveIpAddress": "1.2.3.4",
        "useCellProviderDns": "enabled"

    }
```


```
PUT /modems/ttyM1
        {
            "status": "enabled",
            "profile": "dial_out",
            "phoneNumber": "18001234567",
            "speed": 38400,
            "initChat": "ABORT\tBUSY\nABORT\tVOICE\n",
            "localIpv4Address": "10.20.30.40",
            "remoteIpv4Address": "10.20.30.41",
            "localIpv6Address": "",
            "remoteIpv6Address": "",
            "pppAuthentication": "remote",
            "pppAuthenticationProtocol": "pap",
            "pppRemoteUsername": "testuser",
            "pppRemotePassphrase": "mypassphrase",
            "chapInterval": 0,
            "chapMaxChallenge": 10,
            "chapRestart": 3,
            "pppIdleTimeout": 0
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.7.2 /modems/<NAME>/signalCheck

Use this resource to run a signal quality check on the specified modem. This is only applicable for cellular modems.

Method

POST

Parameters

Parameter	Description
NAME	Name of the modem, such as ttyM1.

Request Body

None.

Response Body

Parameter	Description
status	Signal check status string. Shows the signal quality in terms of both number of bars and a decibel power level.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /modems/ttyM1/signalCheck
```

Response is:

```
{
    "status": "success. 5 of 7 bars, -71 dBm"
}
```

2.7.3 /modems/<NAME>/registration

Use this resource to test registration of the specified modem with the cellular network. This is only applicable for cellular modems.

Methods

POST

Parameters

Parameter	Description
NAME	Name of the modem, such as ttyM1.

Request Body

None.

Response Body

Parameter	Description
status	Registration status string. Shows the results of the attempt to register with a cellular service.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /modems/ttyM1/registration
```

Response is:

```
{
    "status": "success. Registered, Home Network, AT\u0026T, 4G"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.8 CAS Profile

2.8.1 /casProfile

Use this resource to get and set the CAS Profile parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
autoDiscoveryTimeout	Total time in seconds that should be allowed for auto discovery (integer).
probeTimeout	Time in seconds allowed for each individual probe (integer).
defaultDiscoverySpeed	Serial port speed (integer): 300 / 600 / 1200 / 1800 / 2400 / 4800 / 9600 / 19200 / 38400 / 57600 / 115200 / 230400
probe300	enabled / disabled
probe600	enabled / disabled
probe1200	enabled / disabled
probe1800	enabled / disabled
probe2400	enabled / disabled
probe4800	enabled / disabled
probe9600	enabled / disabled
probe19200	enabled / disabled
probe38400	enabled / disabled
probe57600	enabled / disabled
probe115200	enabled / disabled
probe230400	enabled / disabled

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /casProfile

```
{
    "autoDiscoveryTimeout": 60,
    "probeTimeout": 3,
    "defaultDiscoverySpeed": 9600,
    "probe300": "disabled",
    "probe600": "disabled",
    "probe1200": "enabled",
    "probe1800": "disabled",
    "probe2400": "disabled",
    "probe4800": "disabled",
    "probe9600": "enabled",
    "probe19200": "disabled",
    "probe38400": "disabled",
    "probe57600": "enabled",
    "probe115200": "enabled",
    "probe230400": "disabled"
}
```

PUT /casProfile

```
{
    "autoDiscoveryTimeout": 60,
    "probeTimeout": 3,
    "defaultDiscoverySpeed": 9600,
    "probe300": "disabled",
    "probe600": "disabled",
    "probe1200": "enabled",
    "probe1800": "disabled",
    "probe2400": "disabled",
    "probe4800": "disabled",
    "probe9600": "enabled",
    "probe19200": "disabled",
    "probe38400": "disabled",
    "probe57600": "enabled",
    "probe115200": "enabled",
    "probe230400": "disabled"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.8.2 /casProfile/probeStrings[/<INDEX>]

Use this resource to add and remove probe strings used by auto discovery.

Methods

GET, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameters	Description
INDEX	Specifies the integer index of the existing string to access.

Request Body

Parameters	Description
index	Read-only numeric index (integer).
probeString	String to send as a probe to the target device. Example: "\r".

Response Body

Same as request body.

Response Codes

200	OK
201	Created
400	Bad Request

Examples

GET /casProfile/probeStrings

```
{
    "probeStrings": [
        {
            "index": 0,
            "probeString": "\r"
        },
        {
            "index": 1,
            "probeString": "bob"
        }
        …
    ]
}
```

GET /casProfile/probeStrings/0

```
{
        "index": 0,
        "probeString": "\r"
}
```

POST /casProfile/probeStrings

```
{
        "probeString": "test"
}
```

Response is:

```
{
        "index": 2,
        "probeString": "test"
}
```

DELETE /casProfile/probeStrings/2

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.8.3 /casProfile/matchStrings[/<INDEX>]

Use this resource to add and remove match strings used by auto discovery.

Methods

GET, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameters	Description
INDEX	Specifies the integer index of the existing string to access.

Request Body

Parameters	Description
index	Read-only numeric index (integer).
matchString	Regular expression to attempt to match against the response from the target device. Example: "%H!login:".

Response Body

Same as request body.

Response Codes

200	OK
201	Created
400	Bad Request

Examples

GET /casProfile/matchStrings

```
{
    "matchStrings": [
        {
            "index": 0,
            "matchString": "%H.*ogin:"
        },
        {
            "index": 1,
            "matchString": "%H!login:"
        },
        {
            "index": 2,
            "matchString": "%H#"
        },
        {
            "index": 3,
            "matchString": "%H>"
        }
    ]
}
```

GET /casProfile/matchStrings/1

```
{
        "index": 1,
        "matchString": "%H!login:"
}
```

POST /casProfile/matchStrings

```
{
        "matchString": "%H login:"
}
```

Response is:

```
{
        "index": 4,
        "matchString": "%H login:"
}
```

DELETE /casProfile/matchStrings/4

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.8.4 /casProfile/commands[/<NAME>]

Use this resource to add and remove commands used by auto discovery.

Methods

GET, POST, PUT, PATCH, DELETE

Parameters

Fields query is supported for all body parameters.

Parameters	Description
NAME	Specifies the name of the command to access.

Request Body

Parameters	Description
name	Display name of the command.
command	Command line to be executed on the target device after successfully logging in.
match	Regular expression to use to parse the target's response to the command looking for a hostname. Example: "%H"
logoutCommand	Command to send to the target device to logout of a session. If no logoutCommand is specified, the procedure will try logout, then exit, then quit. Multiple commands can be specified separated by commas. Examples: "exit", “logout,quit”
loginPrompt	Regular expression to apply to look for a login prompt of the target device. When the target device returns a response to a newline probe, this regular expression is applied to the response to see if it matches. If no loginPrompt is specified, the default of "[Ll]ogin: " is used. Example: "username:"
passwordPrompt	Regular expression to apply to look for a password prompt of the target device. When the target device returns a response after the console system sends a username, this regular expression is applied to the response to see if it matches. If no passwordPrompt is specified, the default of "[Pp]assword: " is used. Example: "credentials:"
commandPrompt	Regular expression to apply to look for a command prompt of the target device. When the target device returns a response after the console system sends a password, this regular expression is applied to the response to see if it matches a command prompt. If no commandPrompt is specified, the default is “[$#]*”. Example: ">"

Response Body

Same as request body.

Response Codes

200	OK
201	Created
400	Bad Request

Examples

GET /casProfile/commands

```
{
    "casCommands": [
        {
            "name": "hostname",
            "command": "hostname",
            "match": "%H",
            "logoutCommand": "",
            "loginPrompt": "",
            "passwordPrompt": "",
            "commandPrompt": ""
        },
        {
            "name": "uname",
            "command": "uname -a",
            "match": "Linux %H .*$",
            "logoutCommand": "",
            "loginPrompt": "",
            "passwordPrompt": "",
            "commandPrompt": ""
        }
    ]
}
```

GET /casProfile/commands/hostname

```
{
    "name": "hostname",
    "command": "hostname",
    "match": "%H",
    "logoutCommand": "",
    "loginPrompt": "",
    "passwordPrompt": "",
    "commandPrompt": ""
}
```

POST /casProfile/commands

```
{
    "name": "test",
    "command": "testscript.sh",
    "match": "%H"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

Response is:

```
{
    "name": "test",
    "command": "testscript.sh",
    "match": "%H"
}
```

PUT /casProfile/commands/myCommand

```
{
    "command": "/bin/myname -q",
    "match": "%H",
    "logoutCommand": "quit"
}
```

DELETE /casProfile/commands/test

2.8.5 /casProfile/autoAnswer[/<INDEX>]

Use this resource to add and remove auto answer string pairs.

Methods

GET, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameters	Description
INDEX	Specifies the integer index of the existing string pair to access.

Request Body

Parameters	Description
index	Read-only numeric index (integer).
inputString	Input string to compare against incoming data.
outputString	Output string to send in response to matching input data.

Response Body

Same as request body.

Response Codes

200	OK
201	Created
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /casProfile/autoAnswer

```
{
    "autoAnswer": [
        {
            "index": 0,
            "inputString": "myinput",
            "outputString": "myoutput"
        },
        {
            "index": 1,
            "inputString": "yourinput",
            "outputString": "youroutput"
        }
        …
    ]
}
```

GET /casProfile/autoAnswer/0

```
{
    "index": 0,
    "inputString": "myinput",
    "outputString": "myoutput"
}
```

POST /casProfile/autoAnswer

```
{
    "inputString": "testinput",
    "outputString": "testoutput"
}
```

Response is:

```
{
    "index": 2,
    "inputString": "testinput",
    "outputString": "testoutput"
}
```

DELETE /casProfile/autoAnswer/2

2.8.6 /casProfile/poolOfPorts[/<NAME>]

Use this resource to display and edit pool settings, as well as to add new pools and delete existing pools.

Methods

GET, POST, PUT, PATCH, DELETE

Parameters

Fields and filtering queries are supported for all body parameters.

Parameters	Description
NAME	Name of the pool of ports to access.

Request Body

Parameters	Description
name	Pool name.
telnetAliasPort	TCP port used to connect directly to the pool using Telnet protocol (integer).
sshAliasPort	TCP port used to connect directly to the pool using SSH protocol (integer)
rawModeAliasPort	TCP port used to connect directly to the pool using raw socket for connection (integer).
ipv4AliasAddress	IPv4 address used to connect directly to the pool.
ipv4AliasInterface	Interface associated with the pool's ipv4AliasAddress: **eth**#
ipv6AliasAddress	IPv6 address used to connect directly to the pool.
ipv6AliasInterface	Interface associated with the pool's ipv6AliasAddress: **eth**#
members	Comma-separated list of ports that are members of this pool. All members (CAS ports) of the pool must have the same configuration for CAS protocol.

Response Body

Same as request body.

Response Codes

200	OK
201	Created
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /casProfile/poolOfPorts

```
{
    "poolOfPorts": [
        {
            "name": "graphics",
            "telnetAliasPort": 7100,
            "sshAliasPort": 8100,
            "rawModeAliasPort": 0,
            "ipv4AliasAddress": "10.20.30.40",
            "ipv4AliasInterface": "eth0",
            "ipv6AliasAddress": "",
            "ipv6AliasInterface": "eth0",
            "members": "1,2,3"
        },
        {
            "name": "administrative",
            "telnetAliasPort": 7101,
            "sshAliasPort": 8101,
            …
        }
        …
    ]
}
```

GET /casProfile/poolOfPorts/graphics

```
{
    "name": "graphics",
    "telnetAliasPort": 7100,
    "sshAliasPort": 8100,
    …
}
```

POST /casProfile/poolOfPorts

```
{
    "name": "graphics",
    "telnetAliasPort": 7100,
    "sshAliasPort": 8100,
    "members": "1,2,3"
}
```

Response is:

```
{
    "name": "graphics",
    "telnetAliasPort": 7100,
    "sshAliasPort": 8100,
    "rawModeAliasPort": 0,
    "ipv4AliasAddress": "10.20.30.40",
```

```
        "ipv4AliasInterface": "eth0",
        "ipv6AliasAddress": "",
        "ipv6AliasInterface": "eth0",
        "members": "1,2,3"
    }
```

DELETE /casProfile/poolOfPorts/graphics

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.9 Dial-In Profile

2.9.1 /dialinProfile

Use this resource to get and set the Dial-In Profile settings.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
loginToAppliance	Allow dial-in login to appliance: **enabled** / **disabled** / **callback**
otpLoginAuth	OTP login authentication: **enabled** / **disabled**
pppConnection	PPP connection: **enabled** / **disabled** / **callback**
pppPapAuth	PPP/PAP authentication: **dsview_down_local** / **dsview** / **dsview|local** / **kerberos_down_local** / **kerberos** / **kerberos|local** / **ldap_down_local** / **ldap** / **ldap|local** / **local** / **local|ldap** / **local|radius** / **local|tacacs+** / **otp** / **otp|local** / **radius_down_local** / **radius** / **radius|local** / **tacacs+_down_local** / **tacacs+** / **tacacs+|local**
callerIdFilter	Filter incoming calls based on caller ID: **enabled** / **disabled**

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /dialinProfile
    {
        "loginToAppliance": "disabled",
        "otpLoginAuth": "disabled",
        "pppConnection": "disabled",
        "pppAuth": "local",
        "callerIdFilter": "disabled"
    }
```

```
PUT /dialinProfile
    {
        "loginToAppliance": "enabled",
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.9.2 /dialinProfile/callbackUsers[/<NAME>]

Use this resource to add and remove callback users and their numbers from the Dial-In Profile.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of the callback user to access.

Request Body

Parameter	Description
user	Example: "bob"
number	Example: "8001234567"

Response Body

Same as request body.

Response Codes

200	OK
201	Created
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /dialinProfile/callbackUsers
    {
        "callbackUsers": [
            {
                "user": "bob",
                "number": "8001234567"
            },
            {
                "user": "joe",
                "number": "8004567890"
            }
            ...
        ]
    }
```

```
GET /dialinProfile/callbackUsers/bob
    {
        "user": "bob",
        "number": "8001234567"
    }
```

```
PUT /dialinProfile/callbackUsers/bob
    {
        "number": "8001234567"
    }
```

```
POST /dialinProfile/callbackUsers
    {
        "user": "test",
        "number": "8001238378"
    }
```

Response is:

```
    {
        "user": "test",
        "number": "8001238378"
    }
```

```
DELETE /dialinProfile/callbackUsers/test
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.9.3 /dialinProfile/pppOtpUsers[/<NAME>]

Use this resource to add and remove PPP OTP users from the Dial-In Profile.

Methods

GET, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of the PPP OTP user to access.

Request Body

Parameter	Description
user	PPP OTP username. Example: "bob"
passphrase	Write-only value. Example: "thisisaphrase"
seed	Read-only value. Example: "AC9491"

Response Body

Same as request body.

Response Codes

200	OK
201	Created
400	Bad Request

Examples

```
GET /dialinProfile/pppOtpUsers
    {
        "pppOtpUsers": [
            {
                "user": "bob",
                "seed": "AC1234"
            },
            {
                "user": "joe",
                "seed": "AC1240"
            }
            ...
        ]
    }
```

```
GET /dialinProfile/pppOtpUsers/bob
    {
        "user": "bob",
        "seed": "AC1234"
    }
```

```
POST /dialinProfile/pppOtpUsers
    {
        "user": "test",
        "passphrase": "thisisatest"
    }
```

Response is:

```
    {
        "user": "test",
        "seed": "AC6949"
    }
```

```
DELETE /dialinProfile/pppOtpUsers/test
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.9.4 /dialinProfile/callerId[/<INDEX>]

Use this resource to add and remove caller ID numbers used for caller ID filtering.

Methods

GET, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
INDEX	Specifies the integer index of the caller ID number to access.

Request Body

Parameter	Description
index	Read-only numeric index (integer).
number	Example: "8001234567"

Response Body

Same as request body.

Response Codes

200	OK
201	Created
400	Bad Request

Examples

GET /dialinProfile/callerId

```
{
    "numbers": [
        {
            "index": 0,
            "number": "8001234567"
        },
        {
            "index": 1,
            "number": "8001235000-8001235099"
        },
        {
            "index": 2,
            "number": "800123*"
        },
    ]
}
```

GET /dialinProfile/callerId/2

```
{
    "index": 2,
    "number": "800123*"
}
```

POST /dialinProfile/callerId

```
{
    "number": "800123*"
}
```

Response is:

```
{
    "index": 2,
    "number": "800123*"
}
```

DELETE /dialinProfile/callerId/2

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.9.5 /dialinProfile/chapSecrets[/<NAME>]

Use this resource to add and remove CHAP secrets.

Methods

GET, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of the CHAP user to access.

Request Body

Parameter	Description
client	CHAP client username.
secret	CHAP client secret.

Response Body

Same as request body.

Response Codes

200	OK
201	Created
400	Bad Request

Examples

```
GET /dialinProfile/chapSecrets
    {
        "chapSecrets": [
            {
                "client": "bob",
                "secret": "ThisIsASecret"
            },
            {
                "client": "jim",
                "secret": "ThisIsASecretToo"
            },
        ]
    }
```

```
GET /dialinProfile/chapSecrets/bob
    {
        "client": "bob",
        "secret": "ThisIsASecret"
    }
```

```
POST /dialinProfile/chapSecrets
    {
        "client": "jim",
        "secret": "jimsSecret",
    }
```

Response is:

```
    {
        "client": "jim",
        "secret": "jimsSecret"
    }
```

```
DELETE /dialinProfile/chapSecrets/jim
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.10 Pluggable Devices

2.10.1 /pluggableDevices[/<NAME>]

Use this resource to display information about the attached pluggable devices (USB, SD card).

Methods

GET

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Device name of the pluggable device to access.

Request Body

None.

Response Body

Parameter	Description
deviceName	Linux assigned name of the USB device. Example: ttyACM0, ttyUSB0
deviceType	Device type: console / ethernet / modem / storage / wirelessModem
card	Physical device type: "mmc SD" / "usb usbslot"
devicePath	USB device path to uniquely identify where the device is in the USB tree.
deviceInfo	Device info available via USB device descriptors.
status	Current state of the device: inserted/unmounted/ejected
port	Serial port number assigned to this device if it is enabled as a console.

Response Codes

200	OK
400	Bad Request

Examples

GET /pluggableDevices

```
{
    "pluggableDevices": [
        {
            "deviceName": "ttyACM0",
            "deviceType": "console",
            "card": "usb usbslot",
            "devicePath": "1-1.3",
            "deviceInfo": "...",
            "status": "ejected",
            "port": "49"
        },
        ...
    ]
}
```

GET /pluggableDevices/ttyUSB0

```
{
    "deviceName": "ttyUSB0",
    "deviceType": "console",
    "card": "usb usbslot",
    "devicePath": "1-1.4",
    "deviceInfo": "...",
    "status": "inserted",
    "port": "49"
}
```

GET /pluggableDevices/mmcblk1p1

```
{
    "deviceName": "mmcblk1p1",
    "deviceType": "stroage",
    "card": "mmc SD",
    "devicePath": "",
    "deviceInfo": " ",
    "status": "inserted",
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.10.2 /pluggableDevices/<NAME>/setConsole

Use this resource to set up the specified pluggable device as a console port. By adding a new serial port number, the device is added to the list of serial ports. The serial port must then be configured appropriately.

NOTE: This resource only applies to devices that show a deviceType of "console."

Methods

POST

Parameters

Parameter	Description
NAME	Device name of the pluggable device to access.

Request Body

Parameter	Description
port	(Optional) Serial port number to assign to this device if it is enabled as a console. If omitted, the serial port will be allocated according to the USB port location.

Response Body

Parameter	Description
status	Status result, typically “success. <NAME> set to console, port ##” on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /pluggableDevices/ttyUSB0/setConsole
        {
            "status": "success. ttyUSB0 set to console, port 49"
        }
```

```
POST /pluggableDevices/ttyUSB0/setConsole {"port":"58"}
    {
        "status": "success. ttyUSB0 set to console, port 58"
    }
```

2.10.3 /pluggableDevices/<NAME>/eject

Use this resource to eject the specified pluggable device so that it can be physically removed without losing data. For a storage device, this ensures the device is not busy.

Methods

POST

Parameters

Parameter	Description
NAME	Device name of the pluggable device to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /pluggableDevices/sda1/eject
        {
            "status": "It is now safe to physically unplug the sda1 device"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.10.4 /pluggableDevices/<NAME>/delete

Use this resource to delete the specified pluggable device after it has been safely ejected and physically unplugged.

Methods

POST

Parameters

Parameter	Description
NAME	Device name of the pluggable device to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /pluggableDevices/ttyUSB1/delete
        {
            "status": "success. ttyUSB1 deleted"
        }
```

2.11 Authentication

2.11.1 /authentication

Use this resource to get and set the various appliance authentication parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
applianceAuthType	Authentication type to use to grant access to the appliance: **dsview_down_local** / **dsview** / **dsview|local** / **kerberos_down_local** / **kerberos** / **kerberos|local** /**ldap_down_local** / **ldap** / **ldap|local** / **local** / **local|radius** / **local|tacacs+** / **otp** / **otp|local** / **radius_down_local** / **radius** / **radius|local** / **tacacs+_down_local** / **tacacs+** / **tacacs+|local**
fallbackLocalRootConsole	Local root account can access console even when "local" authentication is otherwise not an option: **enabled** / **disabled**
singleSignOn	Single sign-on authentication: **enabled** / **disabled** **NOTE: If enabled, the specified single sign-on authentication is used and no further authentication is needed when accessing a port.**
singleSignOnAuthType	Authentication type to use for single sign-on : **unconfigured** / **dsview_down_local** / **dsview** / **dsview|local**/ **kerberos_down_local** / **kerberos** / **kerberos|local**/ **ldap_down_local** / **ldap** / **ldap|local** / **local** / **local|radius** / **local|tacacs+** / **otp** / **otp|local** / **radius_down_local** / **radius** / **radius|local** / **tacacs+_down_local** / **tacacs+** / **tacacs+|local**
mfa	Multi-Factor Authentication (MFA): **enabled** / **disabled** Enabling MFA adds multi-factor authentication after the main applianceAuthType is authenticated.
mfaSettings:	
type	MFA type: **duo**
applyTo	Authentication service types to which the MFA should be applied: **external** / **local** / **both** **NOTE: The "external" option refers to an external authentication service like Radius versus local authentication done on the appliance itself.**
mfaToken	Include the MFA Token field on the User Login dialog: **enabled** / **disabled** **NOTE: This parameter is not applicable when the mfa setting above is enabled. This is used for displaying an MFA Token field that can be used when a Radius server returns an Access-Challenge message.**

Response Body

Same as request body.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /authentication

```
{
    "applianceAuthType": "local",
    "fallbackLocalRootConsole": "disabled",
    "singleSignOn": "disabled",
    "singleSignOnAuthType": "unconfigured",
    "mfa": "disabled",
    "mfaSettings": {
        "type": "duo",
        "applyTo": "external"
    },
    "mfaToken": "disabled"
}
```

PUT /authentication { "singleSignOn": "enabled" }

2.11.2 /authentication/dsview

Use this resource to get and set the Vertiv™ management software server authentication parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
serverIpAddress1	IP address of the first Vertiv™ management software server.
serverIpAddress2	IP address of the second Vertiv™ management software server.
serverIpAddress3	IP address of the third Vertiv™ management software server.
serverIpAddress4	IP address of the fourth Vertiv™ management software server.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content) for a PUT or PATCH
400	Bad Request

Examples

```
GET /authentication/dsview
        {
            "serverIpAddress1": "10.20.30.40",
            "serverIpAddress2": "0.0.0.0",
            "serverIpAddress3": "0.0.0.0",
            "serverIpAddress4": "0.0.0.0"
        }
```

```
PUT /authentication/dsview { "serverIpAddress1": "10.20.30.40" }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.11.3 /authentication/duo

Use this resource to get and set Duo authentication parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
ikey	The integration key provided by the Duo server administrator.
skey	The secret key provided by the Duo server administrator.
hostname	The API hostname provided by the Duo server administrator.
failmode	The behavior if Duo authentication fails to contact the Duo server: safe / secure. **NOTE: If the Duo server is down or the configuration is incorrect, the "safe" option allows the user to be authenticated after the first factor succeeds. The "secure" option automatically denies the user access if the Duo server cannot be contacted.**

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content) for a PUT or PATCH
400	Bad Request

Examples

```
GET /authentication/duo
    {
        "ikey": "BIQ4LX3DRAPTXGOQZR1P",
        "skey": "",
        "hostname": "api-12345678.duosecurity.com",
        "failmode": "safe"
    }
```

```
PUT /authentication/duo { "hostname": "api-87654321.duosecurity.com" }
```

2.11.4 /authentication/kerberos

Use this resource to get and set Kerberos authentication parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
server	IP address (or realm) of the Kerberos server.
realmDomainName	The Kerberos realm to use.
domainName	The Kerberos domain.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content) for a PUT or PATCH
400	Bad Request

Examples

```
GET /authentication/kerberos
        {
            "server": "10.20.30.41",
            "realmDomainName": "avocent.com",
            "domainName": "avocent.com"
        }
```

```
PUT /authentication/kerberos { "server": "10.20.30.41" }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.11.5 /authentication/ldap

Use this resource to get and set LDAP authentication parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
server	IP address of the LDAP server.
base	Base Distinguished Name to use for ldap operations.
secure	Secure modes: **on** / **off** / **start_tls**
username	Database username.
password	Database password for the username.
attributes	Login attributes.
objectClass	Object Class attribute.
validate	Validate certificate of the remote LDAP server: **enabled** / **disabled**
certificate	Filename of the root certificate to use to validate the remote LDAP server certificate.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /authentication/ldap
    {
        "server": "10.20.30.42",
        "base": "dc=example,dc=com",
        "secure": "off",
        "username": "cn=admin,dc=example,dc=com",
        "password": "mypassword",
        "attributes": "",
        "objectClass": "",
    }
```

```
PUT /authentication/ldap { "server":"10.20.30.42", "secure": "on" }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.11.6 /authentication/radius

Use this resource to get and set RADIUS authentication parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
firstAuthenticationServer	IP address of the first authentication server.
firstAccountingServer	IP address of the first accounting server.
secondAuthenticationServer	IP address of the second authentication server.
secondAccountingServer	IP address of the second accounting server.
secret	Secret word or passphrase, applies to both sets of servers.
timeout	Desired number of seconds for server timeout (integer).
retries	Desired number of retries (integer).
serviceType	Use Radius Service-Type attributes to specify the authorization group: **enabled** / **disabled**
serviceTypeGroups: (only valid if serviceType is enabled)	
login	Authorization group name for Login.
framed	Authorization group name for Framed.
callbackLogin	Authorization group name for Callback Login.
callbackFramed	Authorization group name for Callback Framed.
outbound	Authorization group name for Outbound.
administrative	Authorization group name for Administrative.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /authentication/radius


```
{
    "firstAuthenticationServer": "127.0.0.1",
    "firstAccountingServer": "127.0.0.1",
    "secondAuthenticationServer": "",
    "secondAccountingServer": "",
    "secret": "******",
    "timeout": 3,
    "retries": 2,
    "serviceType": "enabled",
    "serviceTypeGroups": {
        "login": "",
        "framed": "",
        "callbackLogin": "",
        "callbackFramed": "",
        "outbound": "",
        "administrative": ""
    }
}
```


PUT /authentication/radius {"firstAuthenticationServer": "10.20.30.45","timeout":120}

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.11.7 /authentication/tacacs

Use this resource to get and set the TACACS+ authentication parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
firstAuthenticationServer	IP address of the first authentication server.
firstAccountingServer	IP address of the first accounting server.
secondAuthenticationServer	IP address of the second authentication server.
secondAccountingServer	IP address of the second accounting server.
service	Service: ppp / raccess / shell
secret	Secret word or passphrase that applies to both sets of servers.
timeout	Desired number of seconds for server timeout (integer).
retries	Desired number of retries (integer).
version	Version: v0 / v0_v1 / v1 / v1_v0
userLevel	Use TACACS+ User-Level attributes to specify authorization groups: enabled / disabled
userLevelGroups: (only valid if userLevel is enabled)	
userLevel1	Authorization group name for User-Level 1.
...	...
userLevel15	Authorization group name for User-Level 15.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /authentication/tacacs


```
{
    "firstAuthenticationServer": "10.20.30.46",
    "firstAccountingServer": "10.20.30.47",
    "secondAuthenticationServer": "",
    "secondAccountingServer": "",
    "service": "ppp",
    "secret": "",
    "timeout": 10,
    "retries": 2,
    "version": "v1",
    "userLevel": "enabled",
    "userLevelGroups": {
        "userLevel1": "",
        …
        "userLevel15": ""
    }
}
```


PUT /authentication/tacacs {"firstAuthenticationServer": "10.20.30.46", "firstAccountingServer": "10.20.30.47","version": "v0_v1"}

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.12 Users

2.12.1 /users[/<NAME>]

Use this resource to display and edit user settings, as well as to add new users and delete existing users.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of a specific user to access.

Request Body

Parameter	Description
name	Username; admin and root exist by default.
settings	
status	enabled / disabled
password	Password for the user.
changePasswordNextLogin	Set to force the user to change the password the next time they log in: enabled / disabled
userGroups	List of groups to which this user belongs: string array.
passwordExpiration	
minimumDays	Minimum number of days allowed between password changes (integer).
maximumDays	Maximum number of days a password is valid (integer).
inactiveDays	Number of inactive days after which a password is considered expired (integer).
warningDays	Number of days a warning is issued to the user prior to expiration (integer).
accountExpiration	Account expiration date.

Response Body

Same as request body.

Response Codes

200	OK
201	Created
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /users
{
    "users": [
        {
            "name": "admin",
            "settings": {
                "status": "enabled",
                "password": "",
                …
        },
        {
            "name": "root",
            …
        }
    ]
}
```

```
GET /users/admin
{
    "name": "admin",
    "settings": {
        "status": "enabled",
        "password": "",
        "changePasswordNextLogin": "disabled",
        "userGroups": [
            "admin"
        ],
        "passwordExpiration": {
            "minimumDays": 0,
            "maximumDays": 99999,
            "inactiveDays": "",
            "warningDays": 7
        },
        "accountExp": ""
    }
}
```

```
POST /users
{
    "name": "bob",
    "settings": {
        "password": "password1234!",
        "changePasswordNextLogin": "enabled",
        …
}
```

Response is:

```
{
    "name": "bob",
    "settings": {
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
                "status": "enabled",
                "password": "password1234!",
        …
    }
```

2.12.2 /users/<NAME>/unlock

Use this resource to unlock a locked user account.

Methods

POST

Parameters

Parameter	Description
NAME	Name of a specific user to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /users/admin/unlock
```

Response is:

```
{
    "status": "user account unlocked"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.12.3 /users/<NAME>/disable

Use this resource to disable a user account.

Methods

POST

Parameters

Parameter	Description
NAME	Name of a specific user to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /users/root/disable
```

Response is:

```
{
    "status": "user account disabled"
}
```

2.12.4 /users/<NAME>/enable

Use this resource to enable a disabled user account.

Methods

POST

Parameters

Parameter	Description
NAME	Name of a specific user to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /users/root/enable
```

Response is:

```
{
    "status": "user account enabled"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.12.5 /users/passwordRules

Use this resource to configure the appliance password rules parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
checkComplexity	enabled / disabled
complexitySettings:	
minimumDigits	Minimum number of digits (integer).
minimumUppercase	Minimum number of uppercase characters (integer).
minimumSpecial	Minimum number of special characters (integer).
minimumSize	Minimum number of total characters (integer).
expirationSettings:	
minimumDays	Minimum number of days (integer). Default: 0
maximumDays	Maximum number of days (integer). Default: 99999
warningDays	Number of warning days (integer). Default: 7
lockoutSettings:	
failedAttempts	Number of failed attempts allowed before locking account (integer). Default: 10. A value of 0 disables this lockout.
lockoutDuration	Account lockout duration after each failed login in minutes (integer). Default: 0
unlockAfter	Unlock account after a specific number of minutes (integer). Default: 30. A value of 0 means an admin must manually unlock the account.
legacy	Configure to use the legacy password scheme: enabled / disabled Default: disabled

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /users/passwordRules

```
{
    "checkComplexity": "enabled",
    "complexitySettings": {
        "minimumDigits": 0,
        "minimumUppercase": 0,
        "minimumSpecial": 0,
        "minimumSize": 8
    },
    "expirationSettings": {
        "minimumDays": 0,
        "maximumDays": 99999,
        "warningDays": 7,
    },
    "lockoutSettings": {
        "failedAttempts": 10,
        "lockoutDuration": 0,
        "unlockAfter": 30
    },
    "legacy": "disabled"
}
```

PUT /users/passwordRules

```
{
    "checkComplexity": "disabled",
    "lockoutSettings": {
        "failedAttempts":3,
        "lockoutDuration":1,
        "unlockAfter": 5
    }
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.13 Groups

2.13.1 /groups[/<NAME>]

Use this resource to display and edit group settings, as well as to add new users and delete existing users.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of a specific group to access.

Request Body

Parameter	Description
name	Group name. admin, appliance-admin, shell-login-profile and user exist by default.
members	List of users which belong to this group. Array of strings.
settings	
sessionTimeout	global / custom
customTimeout	Custom session timeout in seconds (integer). Default: 300
profile	none / ts_menu / cli
tsMenuOptions	
cliCmd	Example: shell
cliExitAfterExecuting	enabled / disabled

Response Body

Same as request body.

Response Codes

200	OK
201	Created
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /groups

```
{
    "groups": [
        {
            "name": "admin",
            "settings" : {
                "sessionTimeout": "global",
                "customTimeout": 300,
                "profile": "cli",
                "tsMenuOptions": "",
                "cliCmd": "shell",
                "cliExitAfterExecuting": "disabled"
            },
            "members" : [
                "root",
                "admin"
            ]
        },
        {
            "name": "appliance-admin",
            "settings" : {
                …
        }
    ]
}
```

GET /groups/admin

```
{
    "name": "admin",
    "members" : [
        "admin", "root"
    ],
    "settings" : {
        "sessionTimeout": "global",
        "profile": "cli",
        "cliCmd": "shell",
        "cliExitAfterExecuting":"disabled"
    }
}
```

POST /groups {"name":"mygroup"}

Response is:

```
{
    "name": "mygroup"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
PUT /groups/mygroup
        {
            "settings": {
                "sessionTimeout": "custom",
                "customTimeout": 100,
                "profile": "none"
            },
            "members": ["admin","root"]
        }
```

```
PUT /groups/mygroup
        {
            "settings": {
                "sessionTimeout": "custom",
                "customTimeout": 150,
                "profile": "ts_menu",
                "tsMenuOptions": "my options"
            }
        }
```

```
PUT /groups/mygroup
        {
            "settings": {
                "sessionTimeout": "custom",
                "customTimeout": 250,
                "profile": "cli",
                "cliCmd": "shell",
                "cliExitAfterExecuting": "disabled"
            }
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.14 Access Rights

2.14.1 /accessRights/users/<NAME>/serialPorts[/<PORT>]

Use this resource to configure the individual users access rights to serial ports. This is an array of access right structures.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of a specific user to access.
PORT	The integer number of the serial port to access.

Request Body

Parameter	Description
port	Serial port number (integer).
name	Serial port name.
session	single / multi
accessMode	read_write / read_only
killMultiSession	enabled / disabled
sendMessageMultiSession	enabled / disabled
powerControl	enabled / disabled
dataBufferManagement	enabled / disabled
restfulMenu	enabled / disabled

Response Body

Same as request body.

Response Codes

200	OK
201	Created
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /accessRights/users/myuser/serialPorts

```
{
    "accessRights": [
        {
            "port": 1,
            "name": "ttyS1",
            "session": "single",
            "accessMode": "read_write",
            "killMultiSession": "enabled",
            "sendMessageMultiSession": "disabled",
            "powerControl": "enabled",
            "dataBufferManagement": "disabled",

        },
        {
            "port": 2,
            "name": "ttyS2",
            "session": "multi",
            "accessMode": "read_only",
            …
        }
    ]
}
```

GET /accessRights/users/myuser/serialPorts/1

```
{
    "port": 1,
    "name": "ttyS1",
    "session": "single",
    …
}
```

PUT /accessRights/users/myuser/serialPorts/2

```
{
    "session": "read_only"
}
```

POST /accessRights/users/myuser/serialPorts

```
{
    "port": 3,
    "name": "ttyS3",
    "session": "single"
}
```

Response is:

```
{
    "port": 3,
```

```
            "name": "ttyS3",
            "session": "single",
            "accessMode": "read_write",
            "killMultiSession": "disabled ",
            "sendMessageMultiSession": "disabled",
            "powerControl": "disabled ",
            "dataBufferManagement": "disabled",
            "restfulMenu": "disabled",
    }
```

DELETE /accessRights/users/myuser/serialPorts/2

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.14.2 /accessRights/users/<NAME>/appliance

Use this resource to configure the individual users access rights to the appliance.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of a specific user to access.

Request Body

Parameter	Description
view	enabled / disabled
disconnect	enabled / disabled
reboot	enabled / disabled
flashUpgrade	enabled / disabled
configureSettings	enabled / disabled
configureUsers	enabled / disabled
backupRestoreConfig	enabled / disabled
shellAccess	enabled / disabled
transferFiles	enabled / disabled
dialinAccess	enabled / disabled

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /accessRights/users/myuser/appliance
        {
            "view": "disabled",
            "disconnect": "disabled",
            "reboot": "disabled",
            "flashUpgrade": "disabled",
            "configureSettings": "disabled",
            "configureUsers": "disabled",
            "backupRestoreConfig": "disabled",
            "shellAccess": "disabled",
            "transferFiles": "disabled",
            "dialinAccess": "disabled"
        }
```

```
PUT /accessRights/users/myuser/appliance
        {
            "shellAccess": "enabled"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.14.3 /accessRights/users/<NAME>/pdus[/<ID>]

Use this resource to configure the individual users access rights to PDU devices.

Methods

GET, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of a specific group to access.
ID	The pduId of a specific PDU to delete access rights to.

Request Body

Parameter	Description
pduId	PDU name.

Response Body

Same as request body.

Response Codes

200	OK
201	Created
400	Bad Request

Examples

```
GET /accessRights/users/myuser/pdus
        {
            "accessRightsPdu": [
                {
                    "pduId": "MyPDU1"
                },
                {
                    "pduId": "MyPDU2"
                }
            ]
        }
```

```
POST /accessRights/users/myuser/pdus
        {
            "pduId": "newPdu"
        }
```

```
DELETE /accessRights/users/myuser/pdus/newPdu
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.14.4 /accessRights/users/<NAME>/outlets[/<ID>]

Use this resource to configure the individual users access rights to individual PDU outlets.

Methods

GET, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of a specific group to access.
ID	The pduId of a specific PDU to delete outlets access rights.

Request Body

Parameter	Description
pduId	PDU name
outlets	List of outlets

Response Body

Same as request body.

Response Codes

200	OK
201	Created
400	Bad Request

Examples

```
GET /accessRights/users/myuser/outlets
        {
            "accessRightsOutlets": [
                {
                    "pduId": "MyPDU",
                    "outlets": "13-15"
                },
                {
                    "pduId": "MyPDU2",
                    "outlets": "1-3,5"
                }
            ]
        }
```

```
POST /accessRights/users/myuser/outlets
        {
            "pduId": "newPdu",
            "outlets": "7,8,10-12"
        }
```

```
DELETE /accessRights/users/myuser/outlets/newPdu
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.14.5 /accessRights/users/<NAME>/ups[/<ID>]

Use this resource to configure the individual users access rights to UPS devices.

Methods

GET, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of a specific group to access.
ID	The upsId of a specific UPS to delete access rights to.

Request Body

Parameter	Description
upsId	UPS name.

Response Body

Same as request body.

Response Codes

200	OK
201	Created
400	Bad Request

Examples

```
GET /accessRights/users/myuser/ups
        {
            "accessRightsUps": [
                {
                    "upsId": "GXT4_1"
                },
                {
                    "upsId": "GXT4_2"
                }
            ]
        }
```

```
POST /accessRights/users/myuser/ups
        {
            "upsId": "newUps"
        }
```

```
DELETE /accessRights/users/myuser/ups/newUps
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.14.6 /accessRights/users/<NAME>/upsOutletGroups[/<ID>]

Use this resource to configure the individual users access rights to individual UPS outlet groups.

Methods

GET, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of a specific group to access.
ID	The upsId of a specific PS to delete outlets access rights to.

Request Body

Parameter	Description
upsId	UPS name.
outletGroups	List of outlet groups.

Response Body

Same as request body.

Response Codes

200	OK
201	Created
400	Bad Request

Examples

```
GET /accessRights/users/myuser/upsOutletGroups
{
    "accessRightsOutlets": [
        {
            "upsId": "GXT4_1",
            "outletGroups": "1-2"
        },
        {
            "upsId": "GXT4_2",
            "outletGroups": "2"
        }
    ]
}
```

```
POST /accessRights/users/myuser/upsOutletGroups
{
    "upsId": "newUps",
    "outletGroups": "1-2"
}
```

```
DELETE /accessRights/users/myuser/upsOutletGroups/newUps
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.14.7 /accessRights/groups/<NAME>/serialPorts[/<PORT>]

Refer to /accessRights/users/<NAME>/serialPorts[/<PORT>] on page 213.

2.14.8 /accessRights/groups/<NAME>/appliance

Refer to /accessRights/users/<NAME>/appliance on page 216.

2.14.9 /accessRights/groups/<NAME>/pdus[/<ID>]

Refer to /accessRights/users/<NAME>/pdus[/<ID>] on page 218.

2.14.10 /accessRights/groups/<NAME>/outlets[/<ID>]

Refer to /accessRights/users/<NAME>/outlets[/<ID>] on page 220.

2.14.11 /accessRights/groups/<NAME>/ups[/<ID>]

Refer to /accessRights/users/<NAME>/ups[/<ID>] on page 222.

2.14.12 /accessRights/groups/<NAME>/upsOutletGroups[/<ID>]

Refer to /accessRights/users/<NAME>/upsOutletGroups[/<ID>] on page 224.

2.14.13 /accessRights/dsview

Use this resource to configure the access rights to the Vertiv™ management software.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
session	single / multi
accessMode	read_write / read_only
killMultiSession	enabled / disabled
sendMessageMultiSession	enabled / disabled

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /accessRights/dsview
        {
            "session": "single",
            "accessMode": "read_write",
            "killMultiSession": "enabled",
            "sendMessageMultiSession": "disabled",
        }
```

```
PUT /accessRights/dsview
        {
            "accessMode": "read_only"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.15 Events and Logs

2.15.1 /events[/<ID>]

Use this resource to display, enable and disable the various events and control how notifications are sent.

Methods

GET, PUT, PATCH

Parameters

Fields and filtering queries are supported for all body parameters.

Parameter	Description
ID	Event number of the event to access.

Request Body

Parameter	Description
event	Read-only event number (integer).
description	Read-only event description.
traps	Send a trap to one or more SNMP servers: **enabled** / **disabled**
syslog	Send a message to one or more Syslog servers: **enabled** / **disabled**
dsview	Send a message to one or more Vertiv™ management software: **enabled** / **disabled**
email	Send an email: **enabled** / **disabled**
sms	Send a text message: **enabled** / **disabled**

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /events

```
{
    "events":  [
        {
            "event": 1,
            "description": "Appliance Rebooting",
            "traps": "disabled",
            "syslog": "enabled",
            "dsview": "disabled",
            "email": "disabled",
            "sms": "disabled"
        },
        {
            "event": 2,
            "description": "Appliance Daemon Started",
            "traps": "disabled",
            "syslog": "disabled",
            "dsview": "disabled",
            "email": "disabled",
            "sms": "disabled"
        },
        …
    ]
}
```

GET /events/513

```
{
    "events":  513,
    "description": "CPU Temperature Max Threshold Exceeded",
    "traps": "disabled",
    "syslog": "enabled",
    "dsview": "disabled",
    "email": "enabled",
    "sms": "disabled"
}
```

GET /events?syslog=enabled

PUT /events/513

```
{
    "traps": "enabled",
    "syslog": "disabled"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.15.2 /events/syslog

Use this resource to configure the syslog event destination parameters.

NOTE: The two sets of server and port numbers (ipv4 and ipv6) are maintained as ordered lists internally based on the order that the server address is added. As server addresses are changed, deleted, or added the ordering of these entries may change. If editing a particular address/port combination, a GET operation should be done immediately before the PUT/PATCH to ensure that the correct fields are used to edit that server.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
facility	Log Local facility: log_local_0 / log_local_1 / log_local_2 / log_local_3 / log_local_4 / log_local_5
remoteServerIpv4	enabled / disabled
ipv4Server1	Remote server IP address.
ipv4Port1	Remote server UDP port number (integer).
ipv4Server2	Remote server IP address.
ipv4Port2	Remote server UDP port number (integer).
ipv4Server3	Remote server IP address.
ipv4Port3	Remote server UDP port number (integer).
ipv4Server4	Remote server IP address.
ipv4Port4	Remote server UDP port number (integer).
ipv4Server5	Remote server IP address.
ipv4Port5	Remote server UDP port number (integer).
remoteServerIpv6	enabled / disabled
ipv6Server1	Remote server IP address.
ipv6Port1	Remote server UDP port number (integer).
ipv6Server2	Remote server IP address.
ipv6Port2	Remote server UDP port number (integer).
ipv6Server3	Remote server IP address.
ipv6Port3	Remote server UDP port number (integer).
ipv6Server4	Remote server IP address.
ipv6Port4	Remote server UDP port number (integer).

Parameter	Description
ipv6Server5	Remote server IP address.
ipv6Port5	Remote server UDP port number (integer).
applianceConsole	Send syslog messages to the appliance console: **enabled** / **disabled**
rootSession	Send syslog messages to any root sessions: **enabled** / **disabled**

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /events/syslog

```
{
    "facility": "log_local_0",
    "remoteServerIpv4": "enabled",
    "ipv4Server1": "10.20.30.40",
    "ipv4Port1": 514,
    "ipv4Server2": "",
    "ipv4Server3": "",
    "ipv4Server4": "",
    "ipv4Server5": "",
    "remoteServerIpv6": "disabled",
    "applianceConsole": "enabled",
    "rootSession": "disabled"
}
```

PUT /events/syslog

```
{
    "remoteServerIpv6": "enabled",
    "ipv6Server1": "10.20.30.41",
    "ipv6Port1": 514
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.15.3 /events/snmp

Use this resource to configure the event destination SNMP parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
community	Name of the SNMP community.
version	SNMP trap version: v1 / v2 / v3
user	SNMP user.
server1	SNMP server1 IP address.
server2	SNMP server2 IP address.
server3	SNMP server3 IP address.
server4	SNMP server4 IP address.
server5	SNMP server5 IP address.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /events/snmp
        {
            "community": "public",
            "version": "v2",
            "user": "",
            "server1": 10.20.30.40",
            "server2": "",
            "server3": "",
            "server4": "",
            "server5": "",
        }
```

```
PUT /events/snmp
        {
            "server2": 10.20.30.42",
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.15.4 /events/sms

Use this resource to configure the SMS (text message) event destination parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
server	IP address of an SMS server.
port	TCP port number on server (integer).
number	Number to which messages are sent.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /events/sms
        {
            "server": 10.20.30.40",
            "port": 6701,
            "number": "8001234567",
        }
```

```
PUT /events/sms
        {
            "number": "8001234567",
        }
```

2.15.5 /events/email

Use this resource to configure the email event destination parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
server	IP address of an email server.
port	TCP port number on server (integer).
username	Username used to authenticate to the email server if required.
password	Password used to authenticate to the email server if required.
encryption	Method used to encrypt emails if desired: **none** / **ssl** / **tls**. Default: none.
subject	The subject line to include for generated emails. By default, it is blank which means emails will be sent with a subject of "Event Notification".
sender	The email address to use in the "sender" or "from" field of the generated email. By default, it is blank which means emails will be sent with the appliance's hostname as the user and the appliance's domain name as the email server. Example: ACS8048-0520288960@vertiv.com.
destination	Email address to receive event messages.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
GET /events/email
        {
            "server": 10.20.30.40",
            "port": 25,
            "username": "admin@myemail.com",
            "password": "",
            "encryption": "tls",
            "subject": "ACS generated event",
            "sender": "myacsname@mydomain.com",
            "destination": "help@myemail.com,admin@youremail.com"
        }
```

```
PUT /events/email
        {
            "destination": "help@myemail.com"
        }
```

2.15.6 /events/dsview

Use this resource to configure the Vertiv™ management software event destination parameters.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
server	IP address of the Vertiv™ management software.
syslogServerPort	Syslog server port (integer). Default: 4514
sshServerPort	SSH server port (integer). Default: 4122
sshUsername	SSH username.
sshIdleTimeout	SSH idle timeout in seconds (integer). Default: 15
sshStartThreshold	SSH start threshold in bytes (integer). Default: 10000
sshTunnelBufferSize	SSH tunnel buffer size in bytes (integer). Default: 1000000
bufferFullFirstWarning	Buffer full first warning in bytes (integer). Default: 500000
bufferFullSecondWarning	Buffer full second warning in bytes (integer). Default: 700000
bufferFullThirdWarning	Buffer full third warning in bytes (integer). Default: 1000000

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
GET /events/dsview
        {
            "server": "10.20.30.40",
            "syslogServerPort": 4514,
            "sshServerPort": 4122,
            "sshUsername": "admin",
            "sshIdleTimeout": 15,
            "sshStartThreshold": 10000,
            "sshTunnelBufferSize": 1000000,
            "bufferFullFirstWarning": 500000,
            "bufferFullSecondWarning": 700000,
            "bufferFullThirdWarning": 1000000
        }
```

```
PUT /events/dsview
        {
            "server": "10.20.30.40",
            "sshUsername": "admin",
        }
```

2.15.7 /events/trapForward[/<INDEX>]

Use this resource to add and delete trap forward table entries.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
INDEX	Integer index of the trap forward settings to access.

Request Body

Parameter	Description
index	Read-only index (integer).
serverIpAddress	IP address of the server to which traps are sent.
serverUdpPort	UDP port number to which traps are sent (integer).
oid	OID for filter. Default: "default."

Response Body

Same as request body.

Response Codes

200	OK
201	Created
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /events/trapForward

```
{
    "trapForward": [
        {
            "index": 0,
            "serverIpAddress": "10.20.30.40",
            "serverUdpPort": 100,
            "oid": "default"
        },
        {
            "index": 1,
            "serverIpAddress": "10.20.30.41",
            "serverUdpPort": 101,
            "oid": ".1.1.1"
        },
        …
    ]
}
```

GET /events/trapForward/1

```
{
    "index": 1,
    "serverIpAddress": "10.20.30.41",
    "serverUdpPort": 101,
    "oid": ".1.1.1"
}
```

POST /events/trapForward

```
{
    "serverIpAddress": "10.20.30.43",
    "oid": "default"
}
```

Response is:

```
{
    "index": 3,
    "serverIpAddress": "10.20.30.43",
    "serverUdpPort": 162,
    "oid": "default"
}
```

DELETE /events/trapForward/1

2.15.8 /events/trapForward/settings

Use this resource to configure the trap forward settings.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
addForwarderInfo	When a trap is forwarded, add an OID with the IP address of the system from which the trap has been received: yes / no

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /events/trapForward/settings
    {
        "addForwarderInfo": "yes"
    }
```

```
PUT /events/trapForward/settings
    {
        "addForwarderInfo": "yes"
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.15.9 /events/dataBuffering

Use this resource to configure the data buffer settings.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
segmentSize	Local file segment size in Kbytes (integer). 0=unlimited. Default: 256
spareSegments	Number of local spare file segments (integer). Default: 1
nfsServer	IP address of the NFS server to write data.
nfsPath	File path on NFS server to write data.
nfsSegmentSize	NFS file segment size in Kbytes (integer). 0=unlimited. Default: 1024
nfsSpareSegments	Number of nfs spare file segments (integer). Default: 10
closeByTime	Time in HH:MM format when the current log file should be closed and a new one opened.
syslogFacility	Syslog log local facility to use for data buffering: log_local_0 / log_local_1 / log_local_2 / log_local_3 / log_local_4 / log_local_5

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /events/dataBuffering
        {
            "segmentSize": 256,
            "spareSegments": 1,
            "nfsServer": "10.20.30.40",
            "nfsPath": "/mylogging",
            "nfsSegmentSize": 1024,
            "nfsSpareSegments": 10,
            "closeByTime": "01:00",
            "syslogFacility": "log_local_3"
        }
```

```
PUT /events/dataBuffering
        {
            "segmentSize": 64,
            "spareSegments": 5,
            "closeByTime": "23:30",
            "syslogFacility": "log_local_4"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.15.10 /events/applianceLogging

Use this resource to configure appliance logging settings.

Methods

GET, PUT, PATCH

Parameters

Parameter	Description
sessionLogging	Appliance session logging: **enabled** / **disabled**
destination	Destination for writing log file: **local** / **nfs** / **syslog** / **dsview**
localDestination	Local filesystem for logging. Default: mmcblk0
timestamp	Include timestamp in log entries: **enabled** / **disabled**
alerts	Appliance session data logging alerts: **enabled** / **disabled**
alertString1	Regular expression to match for an alert.
alertString2	Regular expression to match for an alert.
alertString3	Regular expression to match for an alert.
alertString4	Regular expression to match for an alert.
alertString5	Regular expression to match for an alert.
alertString6	Regular expression to match for an alert.
alertString7	Regular expression to match for an alert.
alertString8	Regular expression to match for an alert.
alertString9	Regular expression to match for an alert.
alertString10	Regular expression to match for an alert.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /events/applianceLogging
        {
            "sessionLogging": "enabled",
            "destination": "local",
            "localDestination": "mmcblk0",
            "timestamp": "enabled",
            "alerts": "enabled",
            "alertString1": "login",
            "alertString2": "logout",
            …
            "alertString10": "error",
        }
```

```
PUT /events/applianceLogging
        {
            "sessionLogging": "enabled",
            "destination": "nfs"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.15.11 /events/test

Use this resource to test the current events settings.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Status result either “success. test event generated.” or “failure. could not generate test event.”

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

POST /events/test

Response is:

```
{
    "status": "success. test event generated."
}
```

2.16 Power Management

2.16.1 /power/pdus[/<NAME>]

Use this resource to display information about all attached PDUs and to configure PDU settings.

Methods

GET, PATCH, PUT

Parameters

Fields and filtering queries are supported for all body parameters.

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.

Request Body

Parameter	Description
pduId	Name assigned to this specific PDU.
vendor	Read-only name of the vendor of the PDU.
model	Read-only model number of the PDU.
device	Read-only ttyS# form device name, or an IP address for a networked PDU.
number	Read-only Position of the PDU in the chain.
firmwareVersion	Read-only firmware version of the PDU.
maxCurrent	Read-only current rating of the PDU in amps (integer).
numberOfBanks	Read-only number of banks (circuits) in the PDU (integer).
numberOfPhases	Read-only number of phases in the PDU (integer).
numberOfSensors	Read-only number of environmental sensors (integer).
pduIpAddress	Read-only IP address of the PDU, if available.
agentSerialNumber	Read-only serial number of the agent, if available.
pduSerialNumber	Read-only serial number of the PDU, if available.
pduModelNumber	Read-only model number of the PDU, if available.
pduPartNumber	Read-only part number of the PDU, if available.
aggregation	Aggregation feature for Vertiv™ Geist™ rPDUs: **enabled** / **disabled** **NOTE: Only available for the primary rPDU.**
serialPortBaudRate	Read and set the serial port baud rate for a Vertiv™ Geist™ rPDU: **1200** / **2400** / **4800** / **9600** / **19200** / **38400** / **57600** / **115200** **NOTE: Only available for the primary rPDU.**

Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
numberOfOutlets	Read-only number of outlets (integer).
numberOfOutletsOn	Read-only number of outlets which are powered on (integer).
current	
currentValue	Read-only current draw in amps.
currentMax	Read-only maximum observed current in amps.
currentMin	Read-only minimum observed current in amps.
currentAverage	Read-only average observed current in amps.
alarm	Read-only current alarm state of the PDU: normal / low_critical / low_warning / high_critical / high_warning / not_available
highCritical	High critical current threshold in amps.
highWarning	High warning current threshold in amps.
lowWarning	Low warning current threshold in amps.
lowCritical	Low critical current threshold in amps.
voltage	
voltageValue	Read-only voltage in volts.
voltageMax	Read-only maximum observed voltage in volts.
voltageMin	Read-only minimum observed voltage in volts.
voltageAverage	Read-only average observed voltage in volts.
voltageType	Read-only type: measured /estimated
powerConsumption	
powerValue	Read-only power draw in watts.
powerMax	Read-only maximum observed power draw in watts.
powerMin	Read-only minimum observed power draw in watts.
powerAvg	Read-only average observed power draw in watts.
powerType	Read-only type: measured, estimated.
powerFactor	Read-only decimal ratio of the real power to the apparent power.
energyConsumption	
energyValue	Read-only accumulated energy in kWh.
startTime	Read-only time stamp for beginning of energy accumulation.
rackTransferSwitch **NOTE: These options are only supported by the Vertiv™ Geist™ Rack Transfer Switch.**	
preferredSource	Preferred source for the transfer switch: **source_a** / **source_b**
healthTest	Auto health test feature: **enabled** / **disabled**
healthTestPeriod	Number of days (1-365) to wait between health tests.

Parameter	Description
retransfer	Auto retransfer feature: **enabled** / **disabled**
retransferDelay	Delay in seconds (10-360) before an auto-transfer occurs.
lastTransferTime	Read-only date and time when the last transfer occurred.
transferCount	Read-only number of times a transfer has occurred.
powerOutput	Read-only value: true/false **NOTE: If power output is enabled, the system returns "true."**
hardwareFault	Read-only value. :true/false **NOTE: If a hardware fault is detected, the system returns "true."**

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
GET /power/pdus/MPX
```

Response is:


```
{
    "pduId": "MPX",
    "vendor": "Vertiv",
    "model": "MPXPEM-NVAXXAXX",
    "device": "ttyS5",
    "number": "1",
    "firmwareVersion": "14.0.0.2",
    "maxCurrent": 30,
    "numberOfBanks": 3,
    "numberOfPhases": 0,
    "pduIpAddress": "10.207.24.75",
    "numberOfOutlets": 18,
    "numberOfOutletsOn": 18,
    "current": {
        "currentValue": "0.0",
        "currentMax": "0.0",
        "currentMin": "0.0",
        "currentAvg": "0.0",
        "alarm": "low_critical",
        "thresholds": {
            "highCritical": "27.0",
            "highWarning": "15.0",
            "lowWarning": "",
            "lowCritical": "1.8"
            }
        },
        "voltage": {
        "voltageValue": "115",
        "voltageMax": "116",
        "voltageMin": "114",
        "voltageAvg": "114",
        "voltageType": "measured"
        },
        "powerConsumption": {
            "powerValue": "0.0",
            "powerMax": "0.0",
            "powerMin": "0.0",
            "powerAvg": "0.0",
            "powerType": "measured"
        },
        "powerFactor": "0.00",
        "energyConsumption": {
            "energyValue": "0.700",
            "startTime": ""
        }
}
```

```
PUT /power/pdus/MPX
    {
        "current": {
            "thresholds" : {
                "highCritical": "20.0",
                "highWarning": "15",
                "lowCritical": "0",
                "lowWarning": "0"
            }
        }
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.2 /power/pdus/<NAME>/off

Use this resource to turn off all outlets on the PDU.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/off
```

Response is:

```
{
    "status": "success. all outlets turned off."
}
```

2.16.3 /power/pdus/<NAME>/on

Use this resource to turn on all outlets on the PDU.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/on
```

Response is:

```
{
    "status": "success. all outlets turned on."
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.4 /power/pdus/<NAME>/cycle

Use this resource to power cycle all outlets on a PDU.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

POST /power/pdus/MPX/cycle

Response is:

```
{
    "status": "success. all outlets cycled."
}
```

2.16.5 /power/pdus/<NAME>/rename

Use this resource to rename a PDU.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.

Request Body

Parameter	Description
pduId	New name for the PDU.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/rename
        {
            "pduId": "newname"
        }
```

Response is:

```
        {
            "status": "success. PDU renamed newname."
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.6 /power/pdus/<NAME>/resetValues

Use this resource to reset the minimum, maximum, and average electrical monitoring values for a PDU.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.

Request Body

Parameter	Description
valuesType	One of these values: **current** / **voltage** / **power** / **energy**

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/resetValues
        {
            "valuesType":"current"
        }
```

Response is:

```
        {
            "status": "success. reset current for PDU."
        }
```

```
POST /power/pdus/MPX/resetValues
        {
            "valuesType":"voltage"
        }
```

Response is:

```
        {
            "status": "success. reset voltage for PDU."
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.7 /power/pdus/<NAME>/reboot

Use this resource to reboot the PDU controller. This does not affect outlet status.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/reboot
```

Response is:

```
{
    "status": "success. reboot initiated."
}
```

2.16.8 /power/pdus/<NAME>/refresh

Use this resource to refresh the PDU and cause the PDU to be rediscovered. Any attached daisy chained PDUs will also be rediscovered.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/GeistIMD/refresh
```

Response is:

```
{
    "status": "success. refresh initiated"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.9 /power/pdus/<NAME>/factoryDefaults

Use this resource to restore factory defaults on a PDU.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

POST /power/pdus/MPX/factoryDefaults

Response is:

```
{
    "status": "success. Factory defaults initiated"
}
```

2.16.10 /power/pdus/<NAME>/firmwareDownload

Use this resource to download the specified firmware file for the PDU using ftp, in preparation for the subsequent firmware update of the PDU. The action does not return until the file download has completed or fails. Depending on the network speed, this could take a couple of minutes.

NOTE: Not all PDUs support firmware download.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.

Request Body

Parameter	Description
ipAddress	IP address of the remote server from which to download the file.
username	Username to access the remote server.
password	Password to access the remote server.
directory	Directory path on the remote server, typically relative to the ftp root directory.
filename	Filename of the firmware file on the remote server.

Response Body

Parameter	Description
status	Status result.
firmware:	
version	Firmware version number.

Response Codes

200	OK
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
POST /power/pdus/MPX/firmwareDownload
        {
            "ipAddress": "10.20.30.80",
            "username": "anonymous",
            "password": "anonymous",
            "directory": "pub/firmware/",
            "filename": "pdu-firmware.fl"
        }
```

Response is:

```
        {
            "status": "download successful",
            "firmware": {
                "version": "14.0.0.2"
            }
        }
```

2.16.11 /power/pdus/<NAME>/firmwareInstall

Use this resource to install a previously downloaded firmware image into the PDU.

NOTE: This resource does not return until the file has been transferred to the PDU and the upgrade has been initiated, which may take up to two minutes.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.
firmware:	
version	Firmware version number.

Response Codes

200	OK
400	Bad Request

Examples

POST /power/pdus/MPX/firmwareInstall

Response is:

```
{
    "status": "install successful",
    "firmware": {
        "version": "14.0.0.2",
    }
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.12 /power/pdus/<NAME>/outlets[/<INDEX>]

Use this resource to display information about all the outlets on the PDU, as well as to control the outlets and configure the outlet settings.

Methods

GET, PUT, PATCH

Parameters

Fields and filtering queries are supported for all body parameters.

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Integer index of the specific outlet to access.

Request Body

NOTE: All parameters are read-only unless they are marked as read-write.

Parameter	Description
outlet	Outlet numerical index.
name	Outlet name.
status	Outlet status: on / off / on_fixed / on_locked / on_unlocked / on_pend_off / off_pend_on / on_pend_cycle / off_cycle / not_set
bank	Bank number, if applicable. (Not present on all pdus)
current	
currentValue	Current draw in amps.
currentMax	Maximum observed current in amps.
currentMin	Minimum observed current in amps.
currentAverage	Average observed current in amps.
alarm	Current alarm state of the PDU: normal / low_critical / low_warning / high_critical / high_warning / blown_fuse / not_available
highCritical	Read-write high critical current threshold in amps.
highWarning	Read-write high warning current threshold in amps.
lowWarning	Read-write low warning current threshold in amps.
lowCritical	Read-write low critical current threshold in amps.
voltage	
voltageValue	Voltage in volts.
voltageMax	Maximum observed voltage in volts.

Parameter	Description
voltageMin	Minimum observed voltage in volts.
voltageAverage	Average observed voltage in volts.
voltageType	measured or estimated
powerConsumption	
powerValue	Power draw in watts.
powerMax	Maximum observed power draw in watts.
powerMin	Minimum observed power draw in watts.
powerAvg	Average observed power draw in watts.
powerType	measured or estimated
powerFactor	Decimal ratio of the real power to the apparent power.
energyConsumption	
energyValue	Accumulated energy in kWh.
startTime	Timestamp for beginning of energy accumulation.
wakeUpState	Read-write wake state: **last** / **on** / **off**
postOnDelay	Read-write value in seconds.
postOffDelay	Read-write value in seconds.

Response Body

Same as request body.

Examples

GET /power/pdus/MPX/outlets/3

Response is:


```
{
    "number": "3",
    "name": "RCP_A-3_[417271G713C2013AUG01040",
    "bank": "A",
    "status": "on",
    "current": {
        "currentValue": "0.0",
        "currentMax": "0.0",
        "currentMin": "0.0",
        "currentAvg": "0.0",
        "alarm": "normal",
        "thresholds": {
            "highCritical": "19.0",
            "highWarning": "18.0",
            "lowWarning": "",
```

```
                "lowCritical": "0.0"
            }
        },
        "voltage": {
            "voltageValue": "116",
            "voltageMax": "117",
            "voltageMin": "114",
            "voltageAvg": "114",
            "voltageType": "measured"
        },
        "powerConsumption": {
            "powerValue": "0.0",
            "powerMax": "0.0",
            "powerMin": "0.0",
            "powerAvg": "0.0",
            "powerType": "measured"
        },
        "powerFactor": "0.00",
        "energyConsumption": {
            "energyValue": "0.000",
            "startTime": ""
        },
            "wakeUpState": "",
            "postOnDelay": "0.0",
            "postOffDelay": "0.0"
    }
```

```
PUT /power/pdus/MPX/outlets/3
    {
        "name": "newname"
        "postOnDelay": "5"
        "postOffDelay": "5"
        "current": {
            "thresholds": {
                "highCritical": "20.0",
                "highWarning": "15",
                "lowCritical": "0",
                "lowWarning": "0"
            }
        }
    }
```

2.16.13 /power/pdus/<NAME>/outlets/<INDEX>/on

Use this resource to turn on a PDU outlet.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Integer index of the specific outlet to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/outlets/5/on
```

Response is:

```
{
    "status": "success. Outlet 5 turned on"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.14 /power/pdus/<NAME>/outlets/<INDEX>/off

Use this resource to turn off a PDU outlet.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Integer index of the specific outlet to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/outlets/5/off
```

Response is:

```
{
    "status": "success. Outlet 5 turned off."
}
```

2.16.15 /power/pdus/<NAME>/outlets/<INDEX>/cycle

Use this resource to power cycle an outlet. When the operation completes, the outlet will be in the “on” state.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Integer index of the specific outlet to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/outlets/5/cycle
```

Response is:

```
{
    "status": "success. Outlet 5 cycled."
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.16 /power/pdus/<NAME>/outlets/<INDEX>/lock

Use this resource to lock an outlet into its current power state (if supported by the PDU).

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Integer index of the specific outlet to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/outlets/5/lock
```

Response is:

```
{
    "status": "success. outlet 5 locked."
}
```

2.16.17 /power/pdus/<NAME>/outlets/<INDEX>/unlock

Use this resource to unlock an outlet (if supported by the PDU).

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Integer index of the specific outlet to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/outlets/5/unlock
```

Response is:

```
{
    "status": "success. outlet 5 unlocked."
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.18 /power/pdus/<NAME>/outlets/<INDEX>/resetValues

Use this resource to reset the minimum, maximum, and average electrical monitoring values for an outlet.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Integer index of the specific outlet to access.

Request Body

Parameter	Description
valuesType	One of the following values: **current** / **voltage** / **power** / **energy**

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/outlets/1/resetValues
        {
            "valuesType":"current"
         }
```

Response is:

```
        {
            "status": "success. reset current for outlet."
        }
```

2.16.19 /power/pdus/<NAME>/banks[/<INDEX>]

Use this resource to display information about all the banks on the PDU.

Methods

GET, PUT, PATCH

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Integer index of the specific bank to access.

Request Body

NOTE: All parameters are read-only unless they are marked as read-write.

Parameter	Description
number	Number of the bank.
ref	Reference name for the bank.
names	Read-only name for the bank.
bank	Bank number, if applicable.
maxCurrent	Current rating for the bank.
current	
currentValue	Current draw in amps.
currentMax	Maximum observed current in amps.
currentMin	Minimum observed current in amps.
currentAverage	Average observed current in amps.
alarm	Current alarm state of the PDU: normal / low_critical / low_warning / high_critical / high_warning / blown_fuse / not_available
highCritical	Read-write high critical current threshold in amps.
highWarning	Read-write high warning current threshold in amps.
lowWarning	Read-write low warning current threshold in amps.
lowCritical	Read-write low critical current threshold in amps.
voltage	
voltageValue	Voltage in volts.
voltageMax	Maximum observed voltage in volts.
voltageMin	Minimum observed voltage in volts.
voltageAverage	Average observed voltage in volts.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
voltageType	measured or estimated
powerConsumption	
powerValue	Power draw in watts.
powerMax	Maximum observed power draw in watts.
powerMin	Minimum observed power draw in watts.
powerAvg	Average observed power draw in watts.
powerType	measured or estimated
powerFactor	Decimal ratio of the real power to the apparent power.
energyConsumption	
energyValue	Accumulated energy in kWh.
startTime	Time stamp for the beginning of energy accumulation.

Response Body

Same as request body.

Examples

GET /power/pdus/MPX/banks/1


```
{
    "number": "1",
    "name": "BR_A_[417271G713C2013AUG010403]",
    "ref": "A",
     "maxCurrent": "20.0",
    "current": {
        "currentValue": "0.0",
        "currentMax": "0.0",
        "currentMin": "0.0",
        "currentAvg": "0.0",
        "alarm": "low_critical",
        "thresholds": {
            "highCritical": "19.0",
            "highWarning": "18.0",
            "lowWarning": "",
            "lowCritical": "2.0"
        }
    },
    "voltage": {
        "voltageValue": "116",
        "voltageMax": "117",
        "voltageMin": "114",
        "voltageAvg": "114",
        "voltageType": "measured"
    },
    "powerConsumption": {
        "powerValue": "0.0",
        "powerMax": "0.0",
        "powerMin": "0.0",
```

```
            "powerAvg": "0.0",
            "powerType": "measured"
        },
        "powerFactor": "0.00",
        "energyConsumption": {
            "energyValue": "0.000",
            "startTime": ""
          }
        }
```

```
PUT /power/pdus/MPX/banks/3
    {
        "current": {
            "thresholds" : {
                "highCritical": "20.0",
                "highWarning": "15",
                "lowCritical": "0",
                "lowWarning": "0"
            }
        }
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.20 /power/pdus/<NAME>/banks/<INDEX>/resetValues

Use this resource to reset the minimum, maximum, and average electrical monitoring values for a bank.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Integer index of the specific bank to access.

Request Body

Parameter	Description
valuesType	One of the following values: **current** / **voltage** / **power** / **energy**

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/banks/1/resetValues
        {
            "valuesType":"current"
        }
```

Response is:

```
        {
            "status": "success. reset current for bank."
        }
```

2.16.21 /power/pdus/<NAME>/phases[/<INDEX>]

Use this resource to display information about all the phases on the PDU.

Methods

GET, PUT, PATCH

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Number of the phase to access.

Request Body

NOTE: All parameters are read-only unless they are marked as read-write.

Parameter	Description
number	Number of the phase.
ref	Reference name for the phase.
names	Read only name for the phase.
bank	Phase number, if applicable.
maxCurrent	Current rating for the phase.
current	
currentValue	Current draw in amps.
currentMax	Maximum observed current in amps.
currentMin	Minimum observed current in amps.
currentAverage	Average observed current in amps.
alarm	Current alarm state of the PDU: normal / low_critical / low_warning / high_critical / high_warning / blown_fuse / not_available
highCritical	Read-write high critical current threshold in amps.
highWarning	Read-write high warning current threshold in amps.
lowWarning	Read-write low warning current threshold in amps.
lowCritical	Read-write low critical current threshold in amps.
voltage	
voltageValue	Voltage in volts.
voltageMax	Maximum observed voltage in volts.
voltageMin	Minimum observed voltage in volts.

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
voltageAverage	Average observed voltage in volts.
voltageType	measured or estimated
powerConsumption	
powerValue	Power draw in watts.
powerMax	Maximum observed power draw in watts.
powerMin	Minimum observed power draw in watts.
powerAvg	Average observed power draw in watts.
powerType	measured or estimated
powerFactor	Decimal ratio of the real power to the apparent power.
energyConsumption	
energyValue	Accumulated energy in kWh.
startTime	Timestamp for beginning of energy accumulation.

Response Body

Same as request body.

Examples

GET /power/pdus/MPH2/phases

Response is:


```
{
    "number": "1",
    "name": "MPH2",
    "ref": "L1",
    "maxCurrent": "48.0",
    "current": {
        "currentValue": "0.0",
        "currentMax": "0.0",
        "currentMin": "0.0",
        "currentAvg": "0.0",
        "alarm": "normal",
        "thresholds": {
            "highCritical": "43.2",
            "highWarning": "19.2",
            "lowWarning": "",
            "lowCritical": "0.0"
        }
    },
    "voltage": {
        "voltageValue": "208",
        "voltageMax": "208",
```

```
                "voltageMin": "207",
                "voltageAvg": "207",
                "voltageType": "measured"
            },
            "powerConsumption": {
                "powerValue": "0.0",
                "powerMax": "0.0",
                "powerMin": "0.0",
                "powerAvg": "0.0",
                "powerType": "measured"
            },
            "powerFactor": "0.00",
                "energyConsumption": {
                "energyValue": "",
                "startTime": ""
            }
        }
```

```
PUT /power/pdus/MPH2/phases/3
        {
            "current": {
                "thresholds" : {
                    "highCritical": "20.0",
                    "highWarning": "15",
                    "lowCritical": "0",
                    "lowWarning": "0"
                }
            }
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.22 /power/pdus/<NAME>/phases/<INDEX>/resetValues

Use this resource to reset the minimum, maximum, and average electrical monitoring values for a phase.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Number of the phase to access.

Request Body

Parameter	Description
valuesType	One of these values: **current** / **voltage** / **power** / **energy**

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPH2/phases/1/resetValues
        {
            "valuesType":"current"
        }
```

Response is:

```
        {
            "status": "success. reset current for phase."
        }
```

2.16.23 /power/pdus/<NAME>/inlets[/<INDEX>]

Use this resource to display information about all the inlets on the PDU or Vertiv™ Geist™ Rack Transfer Switch.

Methods

GET, PUT, PATCH

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Number of the inlet to access.

Request Body

NOTE: All values are read-only unless they are marked as read-write.

Parameter	Description
number	Number of the inlet.
ref	Reference name for the inlet.
name	Read-only name for the inlet.
current	Current in amps.
voltage	Voltage in volts.
currentCrestFactor	Ratio between peak and RMS current values.
voltageCrestFactor	Ratio between peak and RMS voltage values.
frequency	Line frequency in Hz.
powerQualified	Has the inlet been qualified: true/false **NOTE: Only supported for the Vertiv™ Geist™ Rack Transfer Switch.**
powerActive	Is the inlet active: true/false **NOTE: Only supported for the Vertiv™ Geist™ Rack Transfer Switch.**
alarm	Alarm state of the inlet: normal / low_critical / low_warning / high_critical / high_warning / not_available / source_alarm

Response Body

Same as request body.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
GET /power/pdus/Geist_RTS/inlets
```

Response is:


```
{
    "inlets": [
        {
            "number": "1",
            "name": "Source A",
            "ref": "A",
            "current": "0.2",
            "voltage": "123",
            "currentCrestFactor": "2.7",
            "voltageCrestFactor": "1.3",
            "frequency": "59.9",
            "powerQualified": "true",
            "powerActive": "true",
            "alarm": "normal"
        },
        {
            "number": "2",
            "name": "Source B",
            "ref": "B",
            "current": "0.0",
            "voltage": "0",
            "currentCrestFactor": "1.0",
            "voltageCrestFactor": "1.0",
            "frequency": "0.0",
            "powerQualified": "false",
            "powerActive": "false",
            "alarm": "source_alarm"
        }
    ]
}
```

2.16.24 /power/pdus/<NAME>/sensors[/<INDEX>]

Use this resource to display information about all the sensors on the PDU.

Methods

GET, PUT, PATCH

Parameters

Fields and filtering queries are supported for all body parameters.

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Number of the sensor to access.

Request Body

NOTE: All values are read-only unless they are marked as read-write.

Parameter	Description
number	Number of the sensor.
ref	Reference name for the sensor.
name	Name for the sensor.
unit	Unit value. C,F - temperature sensors % - humidity sensor
sensorValue	Value of sensor.
sensorMax	Maximum observed value.
sensorMin	Minimum observed value.
sensorAverage	Average observed value.
highCritical	Read-write high critical threshold.
highWarning	Read-write high warning threshold.
lowWarning	Read-write low warning threshold.
lowCritical	Read-write low critical current threshold.

Response Body

Same as request body.

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
GET /power/pdus/MPX/sensors
    {
        "number": "1",
        "name": "T-SNSR-P5.1.1",
        "ref": "T1.1",
        "type": "external-temperature",
        "unit": "C",
        "sensorValue": "24.5",
        "sensorMax": "25.0",
        "sensorMin": "24.5",
        "sensorAvg": "24.5",
        "thresholds": {
            "highCritical": "37.0",
            "highWarning": "35.0",
            "lowWarning": "",
            "lowCritical": ""
        }
    }
```

```
PUT /power/pdus/MPX/sensors/1
    {
        "thresholds" : {
            "highCritical": "37.0",
            "highWarning": "35",
            "lowCritical": "0",
            "lowWarning": "0"
        }
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.25 /power/pdus/<NAME>/sensors/<INDEX>/resetValues

Use this resource to reset the minimum, maximum, and average values for a sensor.

Methods

POST

Parameters

Parameter	Description
NAME	Name (pduId) of the specific PDU to access.
INDEX	Number of the sensor to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/pdus/MPX/sensors/1/resetValues
```

Response is:

```
{
    "status": "success. reset values for sensor. "
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.26 /power/ups[/<NAME>]

Use this resource to display information about all attached UPS devices, configure UPS settings and control UPS output and outlet groups.

NOTE: Not all listed parameters are available for all UPS models.

Methods

GET, PATCH, PUT

Parameters

Fields and filtering queries are supported for all body parameters.

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.

Request Body

NOTE: All parameters are read-only unless they are marked as read-write.

Parameter	Description
upsId	Name assigned to this specific UPS.
vendor	Name of the vendor of the UPS.
model	Model number of the UPS.
device	ttyS# form device name, or an IP address for a networked UPS.
input	
inputRmsL1N	System input RMS voltage between line 1 and neutral in VAC.
inputRmsL2N	System input RMS voltage between line 2 and neutral in VAC.
inputRmsL1L2	System input RMS voltage between line 1 and line 2 in VAC.
inputRmsCurrentL1	System input RMS current for line 1 in A AC.
inputRmsCurrentL2	System input RMS current for line 2 in A AC.
inputFrequency	System input frequency in Hz.
inputMaxVoltageL1N	Maximum system input voltage measurement for Line 1-N since the last reset.
inputMinVoltageL1N	Minimum system input voltage measurement for Line 1-N since the last reset.
inputMaxVoltageL2N	Maximum system input voltage measurement for Line 2-N since the last reset.
inputMinVoltageL2N	Minimum system input voltage measurement for Line 2-N since the last reset.
inputNomVoltage	Nominal (or rated) system input voltage in VAC.
inputNomCurrent	Nominal (or rated) system input current in A AC.
inputNomFrequency	Nominal (or rated) system input frequency in Hz.

Parameter	Description
bypass	
bypassInputVoltageL1N	Bypass input RMS voltage between line 1 and neutral in VAC.
bypassInputVoltageL1L2	Bypass input RMS voltage between line 1 and line 2 in VAC.
bypassInputVoltageL2N	Bypass input RMS voltage between line 2 and neutral in VAC.
bypassInputRmsCurrentL1	Bypass input RMS current for line 1 in A AC.
bypassInputRmsCurrentL2	Bypass input RMS current for line 2 in A AC.
bypassInputFrequency	Bypass input frequency in Hz.
bypassNomVoltage	Bypass nominal (or rated) voltage in VAC.
bypassQualificationStatus	Bypass qualification status: normal / fail / marginal_low / marginal_high
bypassModuleOperatingStatus	Operating status for the Bypass Control Module: normal / warning / alarm / fault
bypassNumTransfers	Number of transfers to bypass from inverter since system startup (integer).
battery	
batteryStatus	UPS battery status: normal / low / depleted / unknown
timeRemaining	Calculated time available on battery in minutes.
percentCharge	Percentage of battery charge.
chargeStatus	Battery charge status: fully_charged / not_charging / charging / discharging
dcBusVoltage	Voltage between the positive and negative terminals of the DC bus at the battery input, in VDC.
dcBusNomVoltage	Nominal (or rated) voltage between the positive and negative terminals of the DC bus at the battery input, in VDC.
cabinetType	Type of extended battery cabinets: internal / external / lrt
rating	Total rating of all parallel strings in the battery in AH.
numOfEbcInstalled	Total number of extended battery cabinets installed.
chargeCompensating	Battery charge algorithm changed due to battery temperature: true / false
chargerState	Current state of the battery charger: on / off
nomBatteryCapacity	Nominal (or rated) battery capacity time at full load, in minutes.
dischargeTime	Time on battery operation for this discharge.
floatVoltage	Cell voltage of the battery at float recharging.
testResult	Outcome of the previous battery test: unknown / passed / failed / system_failure / in_progress / unallowed
lowWarningTime	Read-write time in seconds. When battery time remaining falls to, or below, this value, the low battery alarm is activated (integer).
temperature	Highest battery temperature among all installed battery modules, in Celsius
dischargeCycles	Highest number of battery discharge cycles among all installed battery modules (integer).
voltsForCabinet	Voltage between the positive and negative battery terminals of a battery cabinet, in VDC

Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
timeUntilNextTest	Time until the next automatic battery test is started, in minutes
accumDischargeTime	Highest accumulated battery discharge time among installed battery modules, in hours
output	
outputVoltageRmsL1N	System output RMS voltage between Line 1 and Neutral in VAC.
outputVoltageRmsL1L2	System output RMS voltage between Line 1 and Line 2 in VAC.
outputRmsCurrentL1	System output RMS current for Line 1 in A AC.
outputVoltageRmsL2N	System output RMS voltage between Line 2 and Neutral in VAC.
outputRmsCurrentL2	System output RMS current for Line 2 in A AC.
outputFrequency	System output frequency in Hz.
outputMaxVoltageL1N	Maximum system output voltage measurement for Line 1-neutral since last reset.
outputMinVoltageL1N	Minimum system output voltage measurement for Line 1-neutral since last reset.
outputMaxVoltageL2N	Maximum system output voltage measurement for Line 2-neutral since last reset.
outputMinVoltageL2N	Minimum system output voltage measurement for Line 2-neutral since last reset.
outputPower	Sum total of all system output phases.
outputPowerL1	System output power on Line 1.
outputPowerL2	System output power on Line 2.
outputPercentPower	System output power as a percentage of the rated capacity.
outputPercentPowerL1	System output power on Line 1 as a percentage of the rated capacity.
outputPercentPowerL2	System output power on Line 2 as a percentage of the rated capacity.
outputApparentPower	Sum total apparent power of all system output phases.
outputApparentPowerL1	System apparent power on Line 1.
outputApparentPowerL2	System apparent power on Line 2.
outputNomVoltage	Nominal (or rated) system output voltage.
outputApparentPowerRating	Output apparent power rating.
outputNomFrequency	Nominal (or rated) system output frequency.
outputSource	UPS output source: other / normal / off / bypass / battery / booster / reducer
powerFactorCorrection	State of the power factor correction circuitry of the system: on / off
nomPowerFactor	Nominal (or rated) system power factor.
outputOffDelay	Time in seconds to delay before the output shuts off (integer).
UoutputOnDelay	Time in seconds to delay before the system turns on (integer).
outputCycleDelay	Time in seconds to remain off when the output is cycled (integer).
outputQualificationstatus	Output qualification status: normal / fail / marginal_low / marginal_high
ecoMode	

Parameter	Description
ecoModeStatus	Current ECO mode status: enabled / disabled
ecoModeState	Read-write value to enable/disable ECO mode: **enabled** / **disabled**.
systemInfo	
systemStatus	Operating status for the system: normal_operation / startup / normal_with_warning / normal_with_alarm / abnormal_operation
systemModel	System model identifier.
firmwareVersion	Firmware version identifier.
serialNumber	System serial number.
manufactureDate	Manufacturing data of the system.
inputBlackoutCount	Number of occurrences, since the last reset, where the input was not qualified to provide power to the system.
inputBrownoutCount	Number of occurrences, since the last reset, where the system input voltage has fallen below a pre-determined threshold for a specified amount of time.
inverterState	Inverter state: on / off
inletAirTemp	Temperature of the inlet air in degrees Celsius.
shutdownReason	Reason for the most recent shutdown: none / over_temperature / overload / dc_bus_overload / line_swap / low_battery / remote_command / input_inder_voltage / power_factor_correction_fail / external_signal_command
dcConverterStatus	Operating state of the dc converter.
upsTopology	UPS topology: online / offline / line_interactive
bypassInverterInputCfg	Input source configuration for the bypass and inverter: single_combined_source / dual_separate_sources
autoRestart	Read-write value: **enabled** / **disabled** **NOTE: If enabled, the UPS automatically restarts the load when utility power is restored after a battery discharge.**
autoRestartDelay	Read-write value in seconds (integer). **NOTE: If power is lost, the control will delay this amount of time in seconds after power is restored before restarting the unit.**
audibleAlarmControl	Read-write audible alarm control: on / off
events	Return value will be "active" if the event is active, otherwise it will be "normal."
batteryLow	Calculated battery time remaining has reached the low battery threshold.
inputProblem	A problem with the input has been detected.
outputOff	System output is off.
batteryDischarging	Battery is discharging.
replaceBattery	Battery is due for replacement.
batteryTestFail	Battery test failed.
batteryTestInProgress	Battery self test is in progress.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
batteryUndervoltage	Battery voltage is too low.
batteryOvervoltage	Battery voltage has exceeded a predetermined limit.
loadOnBypass	Output power is supplied by the bypass.
badBypass	A problem with the bypass has been detected.
inputUndervoltage	One or more of the input phase voltages had dropped below the limit.
shutdownPending	One or more of the input phase voltages had exceeded the limit.
overTemperature	Equipment over temperature summary event.
outputUndervoltage	One or more of the output phase voltages had dropped below the limit.
outputOvervoltage	One or more of the output phase voltages had exceeded the limit.
outputOverload	An overload exists on the output.
inputBadFrequency	Input frequency is outside the normal range.
chargerFailure	Charger is off.
lossOfRedundancy	Multi-mode collection doesn't have enough modules to satisfy the redundancy configuration.
rectifierFailure	Rectifier is off.
inverterFailure	Inverter output is off.
dctoDcConverterFault	Failure has occurred in the battery discharge circuit.
parallellCommWarning	Parallel communication bus warning.
maintBypassBreaker	Maintenance bypass breaker is closed (if available).
fanFailure	One or more fans have failed (if available).
epoLatched	System output is off - Emergency Power Off latched requires manual reset (if available).
inputWiringFault	The neutral/ground conductors on the input wiring are not properly bonded, or the line/neutral conductors have been swapped (if available).
bypassFrequencyError	Bypass frequency is outside the inverter synchronization limits.
batteryOverTemperature	A battery temperature sensor is reporting a value above a threshold.
inverterShutdown	The inverter has shut down due to a sustained overload.
generalUnspecified	One or more unspecified events are active. See local unit display.
shutdownLowBattery	If active and guaranteed shutdown is enabled, a low battery reserve condition exists that will shutdown the UPS.
powerModuleFailure	One or more power modules are reporting a failure condition.
powerModuleWarning	One or more power modules are reporting a warning condition.
batteryModuleFailure	One or more battery modules are reporting a fault condition.
batteryModuleWarning	One or more battery modules are reporting a warning condition.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /power/ups/GXT4

Response is:


```
{
    "upsId": "GXT4",
    "vendor": "Net-Liebert",
    "model": "MinValidation",
    "device": "10.207.24.67",
    "input": {
        "inputRmsL1N": "113.0",
        "inputRmsCurrentL1": "0.0",
        "inputFrequency": "59.9",
        "inputMaxVoltageL1N": "118.0",
        "inputMinVoltageL1N": "112.0",
        "inputNomVoltage": "120",
        "inputNomCurrent": "8",
        "inputNomFrequency": "60"
    },
    "bypass": {
        "bypassInputVoltageL1N": "113.0",
        "bypassInputRmsCurrentL1": "0.0",
        "bypassInputFrequency": "59.9",
        "bypassNomVoltage": "120"
    },
    "battery": {
        "batteryStatus": "normal",
        "timeRemaining": "178",
        "percentCharge": "100",
        "chargeStatus": "fully_charged",
        "dcBusVoltage": "54",
        "dcBusNomVoltage": "48",
        "cabinetType": "internal",
        "rating": "9",
        "numOfEbcInstalled": "0",
        "chargeCompensating": "false",
        "chargerState": "on",
        "nomBatteryCapacity": "7",
        "dischargeTime": "0",
        "floatVoltage": "54",
        "testResult": "unknown",
        "lowWarningTime": 28
    },
    "output": {
        "outputVoltageRmsL1N": "0.0",
        "outputRmsCurrentL1": "0.0",
```


 Proprietary and Confidential ©2025 Vertiv Group Corp.

```
            "outputFrequency": "0.0",
            "outputMaxVoltageL1N": "117.0",
            "outputMinVoltageL1N": "0.0",
            "outputPower": "0.0",
            "outputPercentPower": "0",
            "outputApparentPower": "0.0",
            "outputNomVoltage": "120",
            "outputApparentPowerRating": "1000",
            "outputNomFrequency": "60",
            "outputSource": "off",
            "powerFactorCorrection": "off",
            "nomPowerFactor": "0.9",
            "outputOffDelay": 10,
            "outputOnDelay": 10,
            "outputCycleDelay": 10
        },
        "ecoMode": {
            "ecoModeStatus": "off",
            "ecoModeState": "enabled"
        },
        "systemInfo": {
            "systemStatus": "normal_with_alarm",
            "systemModel": "MinValidation",
            "firmwareVersion": "U140D170",
            "serialNumber": "1521500545AFB93",
            "manufactureDate": "04AUG15",
            "inputBlackoutCount": "0",
            "inputBrownoutCount": "0",
            "inverterState": "off",
            "inletAirTemp": "20.0",
            "shutdownReason": "remote_command",
            "dcConverterStatus": "off",
            "upsTopology": "online",
            "bypassInverterInputCfg": "single_combined_source",
            "autoRestart": "enabled",
            "autoRestartDelay": 10,
            "audibleAlarmControl": "on"
        },
        "events": {
            "batteryLow": "normal",
            "inputProblem": "normal",
            "outputOff": "active",
            "batteryDischarging": "normal",
            "replaceBattery": "normal",
            "batteryTestFail": "normal",
            "batteryTestInProgress": "normal",
            "batteryUndervoltage": "normal",
            "batteryOvervoltage": "normal",
            "loadOnBypass": "normal",
            "badBypass": "normal",
            "inputUndervoltage": "normal",
            "shutdownPending": "normal",
            "overTemperature": "normal",
            "outputUndervoltage": "normal",
            "outputOvervoltage": "normal",
            "outputOverload": "normal",
            "inputBadFrequency": "normal",
            "chargerFailure": "normal",
```

```
            "lossOfRedundancy": "normal",
            "rectifierFailure": "normal",
            "inverterFailure": "normal",
            "dctoDcConverterFault": "normal",
            "parallellCommWarning": "normal",
            "maintBypassBreaker": "normal",
            "fanFailure": "normal",
            "epoLatched": "normal",
            "inputWiringFault": "normal"
        }
    }
```

```
PUT /power/ups/GXT4
    {
        "battery": {"lowWarningTime" : 30},
        "ecoMode": {"ecoModestate" : "enabled"}
    }
```

```
PUT /power/pdus/GXT4
    {
        "systemInfo": {"autoRestart" : "enabled"},
                      {"autoRestartDelay" : 10"},
                      {"audibleAlarmControl" : "off"}
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.27 /power/ups/<NAME>/rename

Use this resource to rename the specified UPS.

Methods

POST

Parameters

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.

Request Body

Parameter	Description
upsId	New name for the UPS

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/ups/GXT4/rename
        {
            "upsId" : "newname"}
        }
```

Response is:

```
        {
            "status": "success. ups renamed."
        }
```

2.16.28 /power/ups/<NAME>/restoreName

Use this resource to restore the default name for a serially-connected UPS.

NOTE: This does not apply to Net-UPS.

Methods

POST

Parameters

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

POST /power/ups/GXT4/restoreName

Response is:

```
{
    "status": "success. restored ups name."
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.29 /power/ups/<NAME>/outputOn

Use this resource to turn on the UPS output.

Methods

POST

Parameters

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.

Request Body

Request body is optional. If the request body is omitted, the delay defaults to 10 seconds.

Parameter	Description
outputOnDelay	Delay time in seconds (integer).

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/ups/GXT4/outputOn
        {
            "output": {"outputOnDelay" : 10}
        }
```

Response is:

```
        {
            "status": "success. output on ups GXT4 will be turned on, delay 10 seconds."
        }
```

2.16.30 /power/ups/<NAME>/outputOff

Use this resource to turn off the UPS output.

Methods

POST

Parameters

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.

Request Body

Request body is optional. If the request body is omitted, the delay defaults to 10 seconds.

Parameter	Description
outputOffDelay	Delay time in seconds (integer).

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/ups/GXT4/outputOff
        {
            "output": {"outputOffDelay" : 10}
        }
```

Response is:

```
        {
            "status": "success. output on ups GXT4 will be turned off, delay 10 seconds."
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.31 /power/ups/<NAME>/outputCycle

Use this resource to cycle the UPS output off and back on again.

Methods

POST

Parameters

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.

Request Body

Request body is optional. If the request body is omitted, the delay defaults to 10 seconds.

Parameter	Description
outputCycleDelay	Time in seconds that the output will remain off (integer).

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/ups/GXT4/outputCycle
        {
            "output": {"outputCycleDelay" : 10}
        }
```

Response is:

```
        {
            "status": "success. Output on ups GXT4 cycle initiated, delay 10 seconds."
        }
```

2.16.32 /power/ups/<NAME>/testBattery

Use this resource to initiate the battery self test.

Methods

POST

Parameters

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

POST /power/ups/GXT4/testBattery

Response is:

```
{
    "status": "success. Started battery test on ups GXT4"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.33 /power/ups/<NAME>/silenceAlarm

Use this resource to silence the audible alarm.

Methods

POST

Parameters

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

POST /power/ups/GXT4/silenceAlarm

Response is:

```
{
    "status": "success. alarm silenced on ups GXT4"
}
```

2.16.34 /power/ups/<NAME>/resetPowerStats

Use this resource to reset the power statistics.

Methods

POST

Parameters

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

POST /power/ups/GXT4/resetPowerStats

Response is:

```
{
    "status": "success. Power statistics reset on ups GXT4"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.35 /power/ups/<NAME>/outletGroups[/<INDEX>]

Use this resource to display and control the controllable outlet groups on the UPS.

NOTE: Not all UPS devices have controllable outlet groups.

Methods

GET

Parameters

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.
INDEX	Integer index of the specific outlet group to access.

Request Body

None.

Response Body

Parameter	Description
number	Read-only outlet index.
name	Read-only UPS outlet group name.
status	Read-only UPS outlet group status: on / off / invalid / pending_on / pending_off / pending_cycle / pending

Response Codes

200	OK
400	Bad Request

Examples

```
GET /power/ups/GXT4/outletGroups
```

Response is:


```
{
    "outletGroups": [
        {
            "number": "1",
            "name": "GXT4_1",
            "status": "off"
        },
        {
            "number": "2",
            "name": "GXT4_2",
            "status": "off"
        }
    ]
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.36 /power/ups/<NAME>/powerModules

Use this resource to display information about the power modules on the UPS.

NOTE: Not all UPS devices have power modules.

Methods

GET

Parameters

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.

Request Body

None.

Response Body

Parameter	Description
identifier	Read-only index (integer).
operatingStatus	Read-only operating status for the power module: normal / warning / alarm / fault
inverterStatus	Read-only status of the inverter output: active / inactive

Response Codes

200	OK
400	Bad Request

Examples

```
GET /power/ups/APS_UPS/powerModules
```

Response is:


```
{
    "powerModules": [
        {
            "identifier": "1",
            "operatingStatus": "normal",
            "inverterStatus": "active"
        },
        {
            "identifier": "2",
            "operatingStatus": "normal",
            "inverterStatus": "active"
        },
        {
            "identifier": "3",
            "operatingStatus": "normal",
            "inverterStatus": "active"
        },
        {
            "identifier": "4",
            "operatingStatus": "normal",
            "inverterStatus": "active"
        }
    ]
}
```


 Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.37 /power/ups/<NAME>/batteryModules

Use this resource to display information about the battery modules on the UPS.

NOTE: Not all UPS devices have battery modules.

Methods

GET

Parameters

Parameter	Description
NAME	Name (upsId) of the specific UPS to access.

Request Body

None.

Response Body

NOTE: All parameters are read-only.

Parameter	Description
identifier	Numeric index.
operatingStatus	Operating status for the battery module: normal / warning / alarm / fault
batteryStringVoltage	Voltage between the positive and negative battery terminals of a battery string, in VDC.
moduleTemp	Battery temperature (Celsius) measured by the battery module.
numDischargeCycles	Total number of battery discharge cycles for the battery module since it was made.
dischargeTime	Total accumulated discharge time (hours) for the battery module since it was made.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /power/ups/APS_UPS/batteryModules
```

Response is:


```
{
    "batteryModules": [
        {
            "identifier": "9",
            "operatingStatus": "normal",
            "batteryStringVoltage": "84.0",
            "moduleTemp": "25.1",
            "numDischargeCycles": "24",
            "dischargeTime": "11.3"
        },
        {
            "identifier": "10",
            "operatingStatus": "normal",
            "batteryStringVoltage": "83.9",
            "moduleTemp": "25.1",
            "numDischargeCycles": "24",
            "dischargeTime": "11.3"
        },
    ]
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.38 /power/login

Use this resource to configure passwords for the various brands of PDUs.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
avocentPduUsername	Read-only username used to access Avocent-Cyclades PDUs. Default: "admin"
avocentPduPassword	Password used to access Avocent-Cyclades PDUs. Default: "pm8"
spcPduUsername	Read-only username used to access SPC PDUs. Default: "admn"
spcPduPassword	Password used to access SPC PDUs. Default: "admn"
serverTechPduUsername	Read-only username used to access ServerTech PDUs. Default: "admn"
serverTechPduPassword	Password used to access ServerTech PDUs. Default "admn"
raritanPduUsername	Read-only username used to access Raritan PDUs. Default: "admin"
raritanPduPassword	Password used to access Raritan PDUs. Default: "admin"
eatonPduUsername	Read-only username used to access Eaton PDUs. Default: "admin"
eatonPduPassword	Password used to access Eaton PDUs. Default: "admin"
apcPduUsername	Read-only username used to access APC PDUs. Default: "apc"
apcPduPassword	Password used to access APC PDUs. Default: "apc"
vertivPduUsername	Read-only username used to access Vertiv PDUs. Default: "admin"

Parameter	Description
vertivPduPassword	Password used to access Vertiv™ Geist™ rack Power Distrubiton Units (rPDUs). Default: "admin"
geistPduUsername	Username used to access Vertiv™ Geist™ rPDUs. Default: None.
geistPduPassword	Password used to access Vertiv™ Geist™ rPDUs. Default: None.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /power/login
        {
            "avocentCycladesPduUsername": "admin",
            "avocentCycladesPduPassword": "",
            "spcPduUsername": "admn",
            "spcPduPassword": "",
            "serverTechPduUsername": "admn",
            "serverTechPduPassword": "",
            "raritanPduUsername": "admin",
            "raritanPduPassword": "",
            "eatonPduUsername": "admin",
            "eatonPduPassword": "",
            "apcPduUsername": "apc",
            "apcPduPassword": "",
            "vertivPduUsername": "admin",
            "vertivPduPassword": "",
            "geistPduUsername": "",
            "geistPduPassword": ""
        }
```

```
PUT /power/login
        {
            "avocentCycladesPduPassword": "password",
            "geistPduUsername": "test",
            "geistPduPassword": "testpassword"
        }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.39 /power/outletGroups[/<NAME>]

Use this resource to display information about existing outlet groups, create new outlet groups or delete existing outlet groups.

Methods

GET, POST, DELETE

Parameters

Fields and filtering queries are supported for all body parameters.

Parameter	Description
NAME	Name of the specific outlet group to access.

Request Body

Parameter	Description
name	Read-only name of the outlet group.
status	Read-only outlet group status: on / off / on_fixed / on_locked / on_unlocked / on_pend_off / off_pend_on / on_pend_cycle / off_cycle / not_set / unconfigured
outlets	Read-only list of outlets in the group.
powerConsumption	Read-only power draw in watts for the group.

Response Body

Same as request body.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /power/outletGroups
```

Response is:

```
{
    "outletGroups": [
        {
            "name": "group1",
            "status": "on",
            "outlets": "MPX[17-18], PX2[5,7,19-20]",
            "powerConsumption": "0.0"
        },
        {
            "name": "group2",
            "status": "unconfigured",
            "powerConsumption": "N/A"
        },
    ]
}
```

To create a group:

```
POST /power/outletGroups
    {
        "name" : "newGroup"
    }
```

Response is:

```
{
    "name": "newGroup"
}
```

To delete a group:

```
DELETE /power/outletGroups/newGroup
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.40 /power/outletGroups/<NAME>/on

Use this resource to turn on all outlets in the outlet group .

Methods

POST

Parameters

Parameter	Description
NAME	Name of the specific outlet group to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

POST /power/outletGroups/group1/on

Response is:

```
{
    "status": "success. outletgroup group1 turned on."
}
```

2.16.41 /power/outletGroups/<NAME>/off

Use this resource to turn off all outlets in the outlet group.

Methods

POST

Parameters

Parameter	Description
NAME	Name of the specific outlet group to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/outletGroups/group1/off
```

Response is:

```
{
    "status": "success. outletgroup group1 turned off."
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.42 /power/outletGroups/<NAME>/cycle

Use this resource to cycle all outlets in the outlet group off and back on. All outlets will be left in the "on" state.

Methods

POST

Parameters

Parameter	Description
NAME	Name of the specific outlet group to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /power/outletGroups/group1/cycle
```

Response is:

```
{
    "status": "success. outletgroup group1 cycled."
}
```

2.16.43 /power/outletGroups/<NAME>/outlets

Use this resource to display detailed information about the outlets in an outlet group or to add outlets to the group.

Methods

GET, POST

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of the specific outlet group to access.

Request Body

Parameter	Description
id	ID for the outlet consisting of the PDU name and the outlet number in the following format: <pduname>~<outletnumber>
pduId	Name of the PDU for this outlet.
outlet	Outlet number.
status	Outlet status: on / off / on_fixed / on_locked / on_unlocked / on_pend_off / off_pend_on / on_pend_cycle / off_cycle / not_set
powerConsumption	Read-only power draw in watts for the group

Response Body

Same as request body.

Response Codes

200	OK
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
GET /power/outletGroups/group1/outlets
```

Response is:

```
{
    "outlets": [
        {
            "id"   : "PX2~5",
            "pduId": "PX2",
            "outlet": "5",
            "status": "on",
            "powerConsumption": "0.0"
        },
        {
            "id"    : "MPX~5"
            "pduId" : "MPX",
            "outlet": "5",
            "status": "on",
            "powerConsumption": "0.0"
        }
    ]
}
```

To add an outlet (or a range of outlets):

```
POST /power/outletGroups/group1/outlets
    {
        "pduId"  : "PX2",
        "outlet" : "7,12-15"
    }
```

Response is:

```
{
    "pduId" : "PX2",
    "outlet": "7,12-15"
}
```

2.16.44 /power/outletGroups/<NAME>/outlets/<ID>

Use this resource to delete an outlet from an existing outlet group. The ID should consist of the PDU name and the outlet number, in this format: <pduname>~<outletnumber>

Methods

DELETE

Parameters

Parameter	Description
NAME	Name of the specific outlet group to access.
ID	ID of the outlet to access, in the following format: <pduname>~<outletnumber>

Request Body

None.

Response Body

None.

Response Codes

200	OK
400	Bad Request

Examples

```
DELETE /power/outletGroups/group1/outlets/PX2~5
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.16.45 /power/networkPdus[/<IPADDRESS>]

Use this resource to display information about all known Network PDUs, add new Network PDUs, and modify existing entries.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields and filtering queries are supported for all body parameters.

Parameter	Description
IPADDRESS	IP address of the network PDU.

Request Body

Parameter	Description
ipAddress	IP address of the PDU. **NOTE: This parameter is only writeable on a POST to the resource when the entry is created.**
pduType	Type of the network PDU: **net-mph-mpx** / **net-avocent-pmhd** / **net-geist** / **net-servertech** / **net-servertech-pro2** / **net-eaton** / **net-raritan** / **net-apc-rpdu** / **net-apc-rpdu2**
snmpVersion	SNMP version: **v1** / **v2c**
snmpCommunityName	SNMP community name. Example: "public"
pollingRate	Polling rate of the network PDU in seconds (integer).

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /power/networkPdus
{
    "pdus": [
      {
        "ipAddress": "10.207.24.168",
        "pduType": "net-geist",
        "snmpVersion": "v2c",
        "snmpCommunityName": "private",
        "pollingRate": 60
      },
      {
        "ipAddress": "10.207.24.85",
        "pduType": "net-mph",
        "snmpVersion": "v2c",
        "snmpCommunityName": "VertivRackPDU",
        "pollingRate": 60
      }
    ]
}
```

```
PUT /power/networkPdus/10.207.24.63
{
    "snmpVersion": "v2c",
    "snmpCommunityName": "private",
    "pollingRate": 30
}
```

To add a Net-PDU:

```
POST /power/networkPdus
{
    "ipAddress": "10.20.30.42",
    "pduType": "net-geist",
    "snmpVersion": "2c",
    "snmpCommunityName": "public",
    "pollingRate": 60
}
```

Response is:

```
{
    "ipAddress": "10.20.30.40",
    "pduType": "net-geist",
    "snmpVersion": "2c",
    "snmpCommunityName": "public",
    "pollingRate": 60
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

To remove a Net-PDU:

```
DELETE /power/networkPdus/10.207.24.63
```

2.16.46 /power/networkUps[/<IPADDRESS>]

Use this resource to display information about all known Network UPS devices, add new Network UPS device, and modify existing entries.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Parameter	Description
IPADDRESS	IP address of the network UPS.

Request Body

Parameter	Description
ipAddress	IP address of the UPS. **NOTE: This parameter is only writeable on a POST to the resource when the entry is created.**
pduType	Type of the network UPS: **net-liebert**
snmpVersion	SNMP version: **v1** / **v2c**
snmpCommunityName	SNMP community name. Example: "public"
pollingRate	Polling rate of the network UPS in seconds (integer).

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
GET /power/networkUps
{
    "ups": [
      {
        "ipAddress": "10.207.24.67",
        "upsType": "net-liebert",
        "snmpVersion": "v2c",
        "snmpCommunityName": "public",
        "pollingRate": 30
      },
      {
        "ipAddress": "10.207.24.81",
        "upsType": "net-liebert",
        "snmpVersion": "v2c",
        "snmpCommunityName": "public",
        "pollingRate": 60
      }
    ]
  }
```

```
PUT /power/networkUps/10.207.24.673
  {
        "snmpVersion": "v2c",
        "snmpCommunityName": "private",
        "pollingRate": 30
   }
```

To add a Net-UPS:

```
POST /power/networkUps
   {
        "ipAddress": "10.207.24.81",
        "upsType": "net-liebert",
        "snmpVersion": "2c",
        "snmpCommunityName": "public",
        "pollingRate": 60
   }
```

Response is:

```
  {
        "ipAddress": "10.20.24.81",
        "upsType": "net-liebert",
        "snmpVersion": "2c",
        "snmpCommunityName": "public",
        "pollingRate": 60
  }
```

To remove a Net-UPS:

```
DELETE /power/networkUps/10.207.24.81
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.17 Sensors

2.17.1 /sensors/internal

Use this resource to access the internal temperature sensors and voltages of the appliance. Temperatures and voltages are read-only fields, but the minimum, maximum and threshold values can be set for triggering alerts.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Request Body

Parameter	Description
currentCPUTemperature	Current temperature of the CPU in degrees Celsius (integer). Read-only.
maximumCPUTemperature	Maximum CPU Temperature in degrees Celsius (integer).
maximumCPUTemperatureThreshold	Maximum CPU Temperature Threshold in degrees Celsius (integer).
minimumCPUTemperature	Minimum CPU Temperature in degrees Celsius (integer).
minimumCPUTemperatureThreshold	Minimum CPU Temperature Threshold in degrees Celsius (integer).
currentBoardTemperature	Current temperature of the board in degrees Celsius (integer). Read-only.
maximumBoardTemperature	Maximum Board Temperature in degrees Celsius (integer).
maximumBoardTemperatureThreshold	Maximum Board Temperature Threshold in degrees Celsius (integer).
minimumBoardTemperature	Minimum Board Temperature in degrees Celsius (integer).
minimumBoardTemperatureThreshold	Minimum Board Temperature Threshold in degrees Celsius (integer).
voltagePSInternal	PS Internal Supply [0.95v - 1.05v] (float). Read-only.
voltagePLInternal	PL Internal Supply [0.95v ~ 1.05v] (float). Read-only.
voltagePSAuxiliary	PS Auxiliary Supply [1.71v ~ 1.89v] (float). Read-only.
voltagePLAuxiliary	PL Auxiliary Supply [1.71v ~ 1.89v] (float). Read-only.
voltagePSDDR3	PS DDR3 Supply [1.31v ~ 1.39v] (float). Read-only.
voltagePLBlockRam	PL Block RAM Supply [0.95v ~ 1.05v] (float). Read-only.
voltagePowerSupply1	Power Supply 1 voltage [11.06v ~ 12.98v] (float). Read-only.
voltagePowerSupply2	Power Supply 2 voltage [11.06v ~ 12.98v] (float). Read-only (if present).

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

```
GET /sensors/internal
    {
        "currentCPUTemperature": 52,
        "maximumCPUTemperature": 0,
        "maximumCPUTemperatureThreshold": 0,
        "minimumCPUTemperature": 0,
        "minimumCPUTemperatureThreshold": 0
        "currentBoardTemperature": 32,
        "maximumBoardTemperature": 0,
        "maximumBoardTemperatureThreshold": 0,
        "minimumBoardTemperature": 0,
        "minimumBoardTemperatureThreshold": 0
        "voltagePSInternal": 0.97,
        "voltagePLInternal": 0.97,
        "voltagePSAuxiliary": 1.79,
        "voltagePLAuxiliary": 1.79,
        "voltagePSDDR3": 1.34,
        "voltagePLBlockRAM": 0.97,
        "voltagePowerSupply1": 11.66,
        "voltagePowerSupply2": 11.64
    }
```

```
PUT /sensors/internal
    {
        "maximumCPUTemperature": 54,
        "minimumCPUTemperature": 45
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.17.2 /sensors/1Wire[/<ADDRESS>]

Use this resource to access the attached 1-wire sensors.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
ADDRESS	Physical 1-wire address of the sensor to access.

Request Body

Parameter	Description
address	Physical 1-wire address. Read-only.
name	User assignable name of the sensor.
location	User assignable location of the sensor.
type	Type of sensor. Read-only.
value1	Read-only value of sensor element.
value2	Read-only value of sensor element. Contact sensors only.
value3	Read-only value of sensor element. Contact3 sensors only.
maximum	Maximum observed value. Analog sensors only (float). Read-only (if present).
minimum	Minimum observed value. Analog sensors only (float). Read-only (if present).
average	Average observed value. Analog sensors only (float). Read-only (if present).
unit	Sensor units, such as F or C for temperature. Analog sensors only.
thresholdLowWarning	Analog sensors only. Blank for none.
thresholdLowCritical	Analog sensors only. Blank for none.
thresholdHighWarning	Analog sensors only. Blank for none.
thresholdHighCritical	Analog sensors only. Blank for none.
alarm	enabled / disabled **NOTE: Analog sensors only.**
alarm1	disabled / alarm_when_open / alarm_when_closed **NOTE: Contact sensors only.**
alarm2	disabled / alarm_when_open / alarm_when_closed **NOTE: Contact sensors only.**
alarm3	disabled / alarm_when_open / alarm_when_closed

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Parameter	Description
	NOTE: Contact sensors only.
filterTime	Seconds (integer): 1 or a multiple of 15 up to 135 **NOTE: Leak sensors only.**
leakAlarm	enabled / disabled **NOTE: Leak sensors only.**
cableFailAlarm	enabled / disabled **NOTE: Leak sensors only.**

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /sensors/1Wire


```
{
    "1wire": [
        {
            "address": "72390C000000",
            "name": "20.72390C000000",
            "location": "",
            "type": "Dry Contact SN-3C",
            "value1": "sensor_open",
            "value2": "sensor_open",
            "value3": "sensor_closed",
            "alarm1": "alarm_when_closed",
            "alarm2": "alarm_when_open",
            "alarm3": "disabled"
        },
        {
            "address": "E38BB0000000",
            "name": "outside_humidity",
            "location": "outside",
            "type": "External Humidity",
            "value1": "39.03 %",
            "maximum": 43.54,
            "minimum": 38.70,
            "average": 38.94,
            "unit": "%",
            "thresholdLowWarning": "",
            "thresholdLowCritical": "10.5",
            "thresholdHighWarning": "",
            "thresholdHighCritical": "85.0",
```


 Proprietary and Confidential ©2025 Vertiv Group Corp.

```
                "alarm": "disabled"
            },
            {
                "address": "730000003000",
                "name": "leak sensor",
                "location": "closet",
                "type": "Leak SN-L",
                "value1": "normal",
                "filterTime": 30,
                "leakAlarm": "enabled",
                "cableFailAlarm": "disabled"
            },

            ...
        ]
    }
```

```
PUT /sensors/1Wire/72390C000000
    {
        "name": "mydoor",
        "alarm1": "alarm_when_open",
    }
```

To reset the values of a specific sensor (if applicable):

```
DELETE /sensors/1Wire/72390C000000
```

2.17.3 /sensors/1Wire/refresh

Use this resource to refresh the list of sensors by purging the current list and redetecting attached 1-wire devices.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /sensors/1Wire/refresh
```

Response is:

```
{
    "status": "initiated 1-Wire update"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.17.4 /sensors/digitalIn[/<INDEX>]

Use this resource to access the attached Digital In sensors.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Parameter	Description
INDEX	Digital in position to access.

Request Body

Parameter	Description
position	Read-only. Digital In position (integer): 1, 2, 3, or 4
name	User-assigned name of the digital in sensor.
location	User-assigned location of the sensor.
type	Type of sensor.
alarm	Alarm: **disabled** / **alarm_when_open** / **alarm_when_closed**
value	Read-only value of sensor element.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /sensors/digitalIn


```
{
    "digitalIn": [
        {
            "position": 1,
            "name": "1",
            "location": "",
            "type": "vibration",
            "alarm": "disabled",
            "value": "sensor_open"
        },
            "position": 2,
            "name": "1",
            "location": "",
            "type": "vibration",
            "alarm": "disabled",
            "value": "sensor_open"
        },
            "position": 3,
            "name": "1",
            "location": "",
            "type": "vibration",
            "alarm": "disabled",
            "value": "sensor_open"
        },
            "position": 4,
            "name": "mymotionsensor",
            "location": "myoffice",
            "type": "motion_ad_im",
            "alarm": "enabled_when_open",
            "value": "sensor_closed"
        }
    ]
}
```


GET /sensors/digitalIn/1


```
{
    "position": 1,
    "name": "1",
    "location": "",
    "type": "vibration",
    "alarm": "disabled",
    "value": "sensor_open"
}
```


PUT /sensors/digitalIn/2


```
{
    "name": "mydoor",
    "location": "myoffice",
    "alarm": "alarm_when_open"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.17.5 /sensors/pdu

Use this resource to access the sensors attached to PDUs.

Methods

GET

Parameters

Fields query is supported for all body parameters.

Request Body

None.

Response Body

Parameter	Description
name	Read-only name of the sensor.
pdu	Read-only name of the PDU to which the sensor is connected.
type	Type of sensor.
value	Read-only value of sensor element.
maximum	Maximum observed value. Analog sensors only. Read-only (if present).
minimum	Minimum observed value. Analog sensors only. Read-only (if present).
average	Average observed value. Analog sensors only. Read-only (if present).

Response Codes

200	OK
400	Bad Request

Examples

GET /sensors/pdu


```
{
    "sensors": [
        {
            "name": "T-SNSR-RCU-OD.1.1",
            "pdu": "RCU-OD",
            "type": "External Temperature",
            "value": "27.0C (80.0F)",
            "maximum": "27.0C (80.0F)",
            "minimum": "27.0C (80.0F)",
            "average": "27.0C (80.0F)"
        },
        {
            "name": "H-SNSR-RCU-OD.1.2",
            "pdu": "RCU-OD",
            "type": "External Humidity",
            "value": "51.0%",
            "maximum": "51.0%",
            "minimum": "51.0%",
            "average": "51.0%"
        },
        {
            "name": "A-SNSR-RCU-OD.1.3",
            "pdu": "RCU-OD",
            "type": "Air Flow",
            "value": "46",
            "maximum": "46",
            "minimum": "46",
            "average": "46"
        },
        {
            "name": "W-SNSR-RCU-OD.1.4",
            "pdu": "RCU-OD",
            "type": "Dewpoint",
            "value": "16.0C (60.5F)",
            "maximum": "16.0C (60.5F)",
            "minimum": "16.0C (60.5F)",
            "average": "16.0C (60.5F)"
        }
    ]
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.17.6 sensors/pdu/<NAME>/reset

Use this resource to reset the minimum, maximum, and average values for a sensor.

Methods

POST

Parameters

Parameter	Description
NAME	Name of the PDU to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

POST /sensors/pdu/A-SNSR-RCU-OD.1.3/reset

Response is:

```
{
    "status": "success. sensor reset."
}
```

2.18 Digital Out

2.18.1 /digitalOut[/<INDEX>]

Use this resource to access the digital out ports, if present, on the appliance. It allows for naming the ports, reading their current state, or changing the state.

Methods

GET, PUT, PATCH

Parameters

Fields query is supported for all body parameters.

Parameter	Description
INDEX	Digital out position to access.

Request Body

Parameter	Description
position	Read-only. Position of the output (integer): 1 or 2
name	Name assigned to the output.
state	State of the output: on / off

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /digitalOut

```
{
    "digitalOut": [
        {
            "position": 1,
            "name": "lights",
            "state": "on"
        },
        {
            "position": 2,
            "name": "alarm",
            "state": "off"
        }
    ]
}
```

GET /digitalOut/1

```
{
    "position": 1,
    "name": "lights",
    "state": "on"
}
```

PUT /digitalOut/2

```
{
    "state": "on"
}
```

2.19 Monitoring

2.19.1 /monitoring/network/devices[/<INTERFACE>]

Use this resource to access information about the network devices and their current IP addresses.

Methods

GET

Parameters

Fields query is supported for all body parameters.

Parameter	Description
INTERFACE	Name of the network interface. Examples: eth0, eth1, bond0, and so on.

Request Body

None.

Response Body

Parameter	Description
deviceName	Device name: **eth#** / **bond#** / **lte0**
status	Current device status: **enabled** / **disabled** / **plugged** / **unplugged**
ipv4Address	Current device IPv4 address.
ipv4Mask	Current device IPv4 mask.
ipv6Address	Current device IPv6 address.
linkStatus	Current network link status: up / down

Response Codes

200	OK
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

GET /monitoring/network/devices

```
{
    "devices":  [
        {
            "deviceName": "eth0",
            "status": "enabled",
            "ipv4Address": "10.20.30.40",
            "ipv4Mask": "255.255.252.0",
            "ipv6Address": "2001:cdba:0000:0000:0000:0000:5678:1234",
            "linkStatus": "up",
        },                              …
    ]
}
```

GET /monitoring/network/devices/eth2

```
{
    "deviceName": "eth2",
    "status": "enabled",
    "ipv4Address": "10.20.30.42",
    "ipv4Mask": "255.255.255.0",
    "ipv6Address": "",
    "linkStatus": "up"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.19.2 /monitoring/network/routingTables/ipv4

Use this resource to view the ipv4 routing tables.

Methods

GET

Parameters

Fields and filtering queries are supported for all body parameters.

Request Body

None.

Response Body

Parameter	Description
table	Name of the routing table. Multiple routing only.
from	Source prefix to match. Multiple routing only.
destination	The destination network or destination host.
gateway	The gateway address or '*' if none set.
genmask	The netmask for the destination net; '255.255.255.255' for a host destination and '0.0.0.0' for the default route. Not used for multiple routing.
flags	Flags including: U - route is up H – target is a host G – use gateway R – reinstate route for dynamic routing D – dynamically installed by daemon or redirect M – modified from routing daemon or redirect A – installed by addrconf C – cache entry ! – reject route **NOTE: Not used for multiple routing.**
metric	The 'distance' to the target (usually counted in hops). Not used for multiple routing.
ref	Number of references to this route. Not used for multiple routing.
use	Count of lookups for the route. Not used for multiple routing.
source	Source IP address. Multiple routing only.
iface	Network interface to which this route applies, such as eth0, eth1, etc.

Proprietary and Confidential ©2025 Vertiv Group Corp.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /monitoring/network/routingTables/ipv4
    {
        "destination": "default",
        "gateway": "*",
        "genmask": "255.255.252.0",
        "flags": "UG",
        "metric": "0",
        "ref": "0",
        "use": "0",
        "iface": "eth0"
    }
```

When multiple routing tables are enabled (advanced routing):

```
GET /monitoring/network/routingTables/ipv4
    {
        "table": "default",
        "from": "*",
        "destination": "default",
        "gateway": "*",
        "source": "192.168.161.10",
        "iface": "eth0"
    }
```

2.19.3 /monitoring/network/routingTables/ipv6

Use this resource to view the ipv6 routing tables.

Methods

GET

Parameters

Fields and filtering queries are supported for all body parameters.

Request Body

None.

Response Body

Parameter	Description
destination	The destination network or destination host.
nexthop	
flags	Flags including: U - route is up H – target is a host G – use gateway R – reinstate route for dynamic routing D – default, dynamically installed by daemon or redirect M – modified from routing daemon or redirect A – installed by addrconf C – cache entry a – all on link. No routers on link. e – expires n – no next hop f – flow significant route ! – reject route **NOTE: Not used for multiple routing.**
metric	The 'distance' to the target (usually counted in hops). Not used for multiple routing.
ref	Number of references to this route. Not used for multiple routing.
use	Count of lookups for the route. Not used for multiple routing.
iface	Network interface to which this route applies. eth0, eth1, etc.

Response Codes

200	OK
400	Bad Request

Proprietary and Confidential ©2025 Vertiv Group Corp.

Examples

```
GET /monitoring/network/routingTables/ipv6
    {
        "destination": "::/0",
        "nexthop": "::",
        "flags": "UGDAe"
        "metric": "1024",
        "ref": "0",
        "use": "0",
        "iface": "eth0"
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.19.4 /monitoring/serialPorts[/<PORT>]

Use this resource to access information about the current state of the serial ports including transmission counts.

Methods

GET, DELETE

NOTE: The DELETE method is used on an individual serial port to clear the counters to zero.

Parameters

Fields query is supported for all body parameters.

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
port	Serial port number (integer).
deviceName	Serial port device name: Example: ttyS2
name	User-assigned or default name for the port. Default is the last half of the appliance MAC address followed by "-p-" and the port number. Example: 10-2c-3d-p-1
profile	Port profile: cas / power / dial_in / dial_out / socket_client / unconfigured
settings	Physical port settings including speed and parity. Example: 9600 8N1.
target	Target status: on / off / unknown
signals	Current serial signals potentially including: RTS, CTS, DTR, DSR, CD, and RI
txBytes	Number of transmitted bytes (integer).
rxBytes	Number of received bytes (integer).
frameErrors	Number of frame errors (integer).
parityErrors	Number of parity errors (integer).
break	Number of breaks (integer).
overrun	Number of overruns (integer).

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Response Codes

200	OK
400	Bad Request

Examples

GET /monitoring/serialPorts


```
{
    "monitorSerialPorts": [
        {
            "port": 1,
            "deviceName": "ttyS1",
            "profile": "cas",
            "settings": "9600 8N1",
            "target": "on",
            "signals": "CTS|DSR|CD|RI",
            "txBytes": 0,
            "rxBytes": 967,
            "frameErrors": 0,
            "parityErrors": 0,
            "break": 0,
            "overrun": 0
        },
        {
            "port": 2,
            "deviceName": "ttyS2",
            "profile": "power",
            "settings": "9600 8N1",
            "target": "on",
            "signals": "RTS|CTS|DTR|DSR|CD|RI",
            "txBytes": 21,
            "rxBytes": 510,
            "frameErrors": 0,
            "parityErrors": 0,
            "break": 0,
            "overrun": 0
        },
        …
    ]
}
```


GET /monitoring/serialPorts/1


```
{
    "port": 1,
    "deviceName": "ttyS1",
    "profile": "cas",
    "settings": "9600 8N1",
    "signals": "CTS|DSR|CD|RI",
    "txBytes": 0,
    "rxBytes": 967,
    "frameErrors": 0,
    "parityErrors": 0,
    "break": 0,
```

```
    "overrun": 0
}
```

DELETE /monitoring/serialPorts/2

Proprietary and Confidential ©2025 Vertiv Group Corp.

2.19.5 /monitoring/scheduledTasks[/<NAME>]

Use this resource to create scheduled tasks to be run on the appliance and monitor existing tasks.

Methods

GET, PUT, PATCH, POST, DELETE

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of the scheduled task to access.

Request Body

Parameter	Description
name	Scheduled task name.
type	Type of task whether a built-in task or a custom task: cell_modem_ip_test / cell_modem_signal_check / save_config_cli / custom
status	Task status: enabled/disabled
frequency	How often the task is executed: hourly/daily/weekly/monthly
day	If weekly, the day of the week: sunday/monday/.../saturday
date	If monthly, the day of the month: 1-31
time	The time of day in HH:MM format. NOTE: **If hourly, the HH portion is ignored.**
script	The name and path of the script to run and the arguments to pass to it.
ipAddress	An argument for certain task types such as cell_modem_ip_test.
lastRuntime	The date and time that the task was last executed or blank.
lastResult	The result of the task the last time it was executed or “Not attempted”.

Response Body

Same as request body.

Response Codes

200	OK
204	OK No Content (for a PUT or PATCH)
400	Bad Request

Examples

GET /monitoring/scheduledTasks


```
{
    "scheduledTasks": [
        {
            "name": "MySigCheck",
            "type": "cell_modem_signal_check",
            "status": "enabled",
            "frequency": "daily",
            "day": "sunday",
            "date": "1",
            "time": "01:30",
            "script": "",
            "lastRuntime": "2021-04-22 01:30:01",
            "lastResult": "Succeeded: -61db, BitErrRate 3.2% to 6.4%"
        },
        {
            "name": "CustomExample",
            "type": "custom",
            "status": "enabled",
            "frequency": "weekly",
            "day": "monday",
            "date": "1",
            "time": "12:30",
            "script": "example.sh",
            "lastRuntime": "2021-04-22 12:30:01",
            "lastResult": "Succeeded: 210422 12:30"
        },
        {
            "name": "iptest",
            "type": "cell_modem_ip_test",
            "status": "enabled",
            "frequency": "daily",
            "day": "sunday",
            "date": "1",
            "time": "18:50",
            "script": "",
            "lastRuntime": "2021-04-22 18:50:01",
            "lastResult": "Succeeded",
            "ipAddress": "64.233.177.101"
        },
        …
    ]
}
```


GET /monitoring/scheduledTasks/MySigCheck


```
{
    "name": "MySigCheck",
    "type": "cell_modem_signal_check",
    "status": "enabled",
    "frequency": "daily",
    "day": "sunday",
    "date": "1",
    "time": "01:30",
```

```
        "script": "",
        "lastRuntime": "2021-04-22 01:30:01",
        "lastResult": "Succeeded: -61db, BitErrRate 3.2% to 6.4%"
    }
```

```
GET /monitoring/scheduledTasks/MySigCheck?fields=lastRuntime,lastResult
    {
        "name": "MySigCheck",
        "lastRuntime": "2021-04-22 01:30:01",
        "lastResult": "Succeeded: -61db, BitErrRate 3.2% to 6.4%"
    }
```

```
PUT /monitor/scheduledTasks/MySigCheck
    {
        "frequency": "weekly",
        "day": "tuesday",
    }
```

```
POST /monitor/scheduledTasks
    {
        "name": "MySigCheck",
        "type": "cell_modem_signal_check",
        "status": "disabled",
        "frequency": "daily",
        "time": "01:30",
    }
```

Response is:

```
    {
        "name": "MySigCheck",
        "type": "cell_modem_signal_check",
        "status": "disabled",
        "frequency": "daily",
        "time": "01:30"
    }
```

```
DELETE /monitoring/scheduledTasks/MySigCheck
```

2.19.6 /monitoring/scheduledTasks/<NAME>/runNow

Use this resource to run the specified task immediately.

Methods

POST

Parameters

Parameter	Description
NAME	Name of the scheduled task to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /monitoring/scheduledTasks/MyTask/runNow
```

Response is:

```
{
    "status": "success. Running task MyTask."
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.19.7 /monitoring/ipsec[/<NAME>]

Use this resource to access information about the status of the IPsec connections and tunnel status.

Methods

GET

Parameters

Fields query is supported for all body parameters.

Parameter	Description
NAME	Name of the IPSec connection to access.

Request Body

None.

Response Body

Parameter	Description
connectionName	IPSec connection name.
status	Connection status: down / connecting / established
remoteIpAddress	IP address of the remote side.
lifetime	Connection's lifetime.
establishedTime	How long the tunnel has been established.
phase1Algorithm	Algorithm negotiated during phase 1.
phase2Algorithm	Algorithm negotiated during phase 1.
certificateName	Name of the certificate use for authentication if appropriate.

Response Codes

200	OK
400	Bad Request

Examples

GET /monitoring/ipsec

```
{
    "ipsecConnections": [
        {
            "connectionName": "testConnection",
            "status": "down",
            "remoteIpAddress": "10.20.30.40",
            "lifetime": "24h",
            "establishedTime": "",
            "phase1Algorithm": "",
            "phase2Algorithm": "",
            "certificateName": ""
        },
        …
    ]
}
```

GET /monitoring/ipsec/testConnection

```
{
    "connectionName": "testConnection",
    "status": "down",
    "remoteIpAddress": "10.20.30.40",
    "lifetime": "24h",
    "establishedTime": "",
    "phase1Algorithm": "",
    "phase2Algorithm": "",
    "certificateName": ""
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.19.8 /monitoring/autoDiscovery[/<PORT>]

Use this resource to monitor and debug the auto discovery process for detecting the names of devices attached to ports.

Methods

GET

Parameters

Fields query is supported for all body parameters.

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
port	Serial port number (integer).
type	Type of auto discovery: probe / command
status	Current status of the discovery process: off / idle / running / failed / succeeded
lastRuntime	Date and time that auto discovery was last executed or blank.
name	Name of the device detected, if any.
logfile	Log output of the last auto discovery attempt for this port.

Response Codes

200	OK
400	Bad Request

Examples

GET /monitoring/autoDiscovery


```
{
    "autoDiscoveryStatus": [
        {
            "port": 2,
            "type": "command",
            "status": "running",
            “lastRuntime”: “2024-09-25 14:00:33”,
            "name": "MyLinuxServer",
            "logfile": "14:55:38 AD Port 2 Autodiscovery started, type=Command, command=\n14:55:38 AD Port 2 looking
for login prompt\n14:55:38 AD Port 2 newline sent\n14:55:39 AD
            Port 2 data processed: … "
        },
        {
            "port": 3,
            "type": "probe",
            "status": "succeeded",
            “lastRuntime”: “2024-09-25 13:33:54”,
            "name": "Cisco1911",
            "logfile": "01:37:47 AD Port 3 Autodiscovery started, type=Probe\n01:37:47 AD Port 3 probe string sent: \\r\n01:
37:49 AD Port 3 data processed: WRLCisco1911\u003e\n01:37
            :49 AD Port 3 probe string sent: \\r\n01:37:49 AD Port 3 data processed: WRLCisco1911\\nWRLCisco1911\u00
3e\n01:37:49 AD Port 3 match string detected: %H\u003e\n01:
            37:49 AD Port 3 Autodiscovery completed successfully [name: WRLCisco1911]\n"
        },
    ]
}
```


GET /monitoring/autoDiscovery/3?fields=name


```
{
    "port": 3,
    "name": "Cisco1911"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.19.9 /monitoring/autoDiscovery/<PORT>/discoverNow

Use this resource to initiate the Auto Discovery procedure to determine the port name for an auto discovery port.

Methods

POST

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result, typically “success. port <PORT> auto discovery initiated." on success.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /monitoring/autoDiscovery/3/discoverNow
```

Response is:

```
{
    "status": "success. port 3 auto discovery initiated."
}
```

2.19.10 /monitoring/fipsMode

Use this resource to monitor the services in FIPS mode.

Methods

GET

Parameters

Fields and filtering queries are supported for all body parameters.

Request Body

None.

Response Body

Parameter	Description
serviceName	Name of the service: HTTPS / SNMPv3 / SSH
modeIndication	Indication that the service is in FIPS mode: fipsMode / nonFipsMode

Response Codes

200	OK
400	Bad Request

Examples

GET /monitoring/fipsMode

```
{
    "fipsModeServices": [
        {
            "serviceName": "SSH",
            "modeIndication": "fipsMode"
        },
        {
            "serviceName": "SNMPv3",
            "modeIndication": "fipsMode"
        },
        {
            "serviceName": "HTTPS",
            "modeIndication": "fipsMode"
        }
    ]
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.19.11 /monitoring/callerIdLog

Use this resource to display the caller ID log.

Methods

GET

Parameters

Fields and filtering queries are supported for all body parameters.

Request Body

None.

Response Body

Array of caller ID log entries:

Parameter	Description
timestamp	Timestamp of the call.
number	Phone number of the caller.
action	Action taken: answered / blocked
name	Name displayed for the caller.

Response Codes

200	OK
400	Bad Request

Examples

GET /monitoring/callerIdLog


```
{
    "callerIdLogEntries": [
        {
            "timestamp": "2019 Jul 10 14:59:18",
            "number": "7153868861",
            "action": "answered",
            "name": "AVOCENT"
        },
        {
            "timestamp": "2019 Jul 10 15:41:27",
            "number": "7153868861",
            "action": "answered",
            "name": "AVOCENT"
        },
    ]
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.19.12 /monitoring/callerIdLog/clearLog

Use this resource to clear the caller ID log.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /monitoring/callerIdLog/clearLog
```

Response is:

```
{
    "status": "success. Caller ID log cleared."
}
```

2.19.13 /monitoring/modemPppdLog

Use this resource to display the Modem PPPD (point-to-point protocol daemon) log. If the log is very large, then only the newest entries will be displayed.

Methods

GET

Parameters

Fields query is supported for all body parameters.

Request Body

None.

Response Body

Array of modem pppd log entries:

Parameter	Description
timestamp	Timestamp of the log entry.
level	Log message level, such as info, notice, error, etc.
logEntry	Text of the Modem PPPD log entry.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /monitoring/modemPppdLog
    {
        "modemPppdLogEntries": [
            {
                "timestamp": "Nov  4 16:55:00",
                "level": "<notice>",
                "logEntry": "pppd[13396]: pppd 2.4.8 started by root, uid 0"
            },
            {
                "timestamp": "Nov  4 16:55:01",
                "level": "<info>",
                "logEntry": "chat[13404]: abort on (NO DIAL TONE)"
            }
        ]
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.19.14 /monitoring/modemPppdLog/clearLog

Use this resource to clear the Modem PPPD log.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /monitoring/modemPppdLog/clearLog
```

Response is:

```
{
    "status": "success. PPPD Log cleared."
}
```

2.19.15 /monitoring/zeroTouchLog

Use this resource to display the Zero Touch log. If the log is very large, then only the newest entries will be displayed.

Methods

GET

Parameters

Fields query is supported for all body parameters.

Request Body

None.

Response Body

Array of zero touch log entries:

Parameter	Description
timestamp	Timestamp of the log entry.
logEntry	Text of the Zero Touch log entry.

Response Codes

200	OK
400	Bad Request

Examples

```
GET /monitoring/zeroTouchLog
    {
        "zeroTouchLogEntries": [
            {
                "timestamp": "Fri Nov 3 19:23:12 UTC 2020",
                "logEntry": "Downloaded SETUP file: ACS_157856.img "
            },
            {
                "timestamp": "Fri Nov 3 19:23:12 UTC 2020",
                "logEntry": "Current firmware version: 1.5.16. Required version: 1.5.18 "
            }
        ]
    }
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.19.16 /monitoring/zeroTouchLog/clearLog

Use this resource to clear the Zero Touch log.

Methods

POST

Parameters

None.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /monitoring/zeroTouchLog/clearLog
```

Response is:

```
{
    "status": "success. Zero Touch log cleared."
}
```

2.20 Access

2.20.1 /access/serialPorts[/<PORT>]

Use this resource to view a list of all serial ports with a CAS profile setting.

Methods

GET

Parameters

Fields and filtering queries are supported for all body parameters.

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
port	Read-only port number.
name	Read-only name of the serial port.
target	Read-only target status: on / off / unknown
status	Read-only status of the serial port: idle / in_use

Response Codes

200	OK
4010	Bad Request
401	Not Authorized

Proprietary and Confidential ©2025 Vertiv Group Corp.

Example

```
GET /access/serialPorts/1
```

Response is:

```
{
    "port": "1",
    "name": "port1",
    "target": "on",
    "status": "idle"
}
```

2.20.2 /access/serialPorts/<PORT>/on

Use this resource to turn on the power outlets merged with this port.

Methods

POST

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /access/serialPorts/1/on
```

Response is:

```
{
    "status": "success. target powered on"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.20.3 /access/serialPorts/<PORT>/off

Use this resource to turn off the power outlets merged with this port.

Methods

POST

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /access/serialPorts/1/off
```

Response is:

```
{
    "status": "success. target powered off"
}
```

2.20.4 /access/serialPorts/<PORT>/cycle

Use this resource to cycle power on the power outlets merged with this port.

Methods

POST

Parameters

Parameter	Description
PORT	The integer number of the serial port to access.

Request Body

None.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

POST /access/serialPorts/1/cycle

Response is:

```
{
    "status": "success. target power cycled"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.21 Miscellaneous

2.21.1 /changePassword

Use this resource to change the password of the authenticated local RestAPI user or set the password on a new Avocent ACS800/8000 advanced console system where the admin password is initially blank, and the user is required to change the password before logging in.

NOTE: This resource is only valid with the Basic Authentication method and is the ONLY RestAPI resource accessible with a blank password.

After the password is set with this action, then the user can login normally via JWT Authentication or continue to use Basic Authentication with the new password for future RestAPI requests.

Methods

POST

Parameters

None.

Request Body

Parameter	Description
password	New password for the currently authenticated RestAPI user.

Response Body

Parameter	Description
status	Status result.

Response Codes

200	OK
400	Bad Request

Examples

```
POST /changePassword {"password":"mypassword"}
```

Response is:

```
{
    "status": "password successfully changed for admin"
}
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

2.21.2 /resources

Use this resource to view a list of the available resources and methods.

Methods

GET

Parameters

None.

Request Body

None.

Response Body

Array of resource methods and URLs:

Parameter	Description
resource	String showing method and URL including path parameters.

Response Codes

200	OK
400	Bad Request

Examples

GET /resources

```
{
    "resources": [
        "GET        /access/serialPorts",
        "GET        /access/serialPorts/:PORT",
        "POST       /access/serialPorts/:PORT/cycle",
        "POST       /access/serialPorts/:PORT/off",
        "POST       /access/serialPorts/:PORT/on",
        "GET        /accessRights/:BASE/:NAME/appliance",
        "PATCH      /accessRights/:BASE/:NAME/appliance",
        "PUT        /accessRights/:BASE/:NAME/appliance",
        "GET        /accessRights/:BASE/:NAME/outlets",
        "POST       /accessRights/:BASE/:NAME/outlets",
        "DELETE     /accessRights/:BASE/:NAME/outlets/:ID",
        "GET        /accessRights/:BASE/:NAME/pdus",
        "POST       /accessRights/:BASE/:NAME/pdus",
        "DELETE     /accessRights/:BASE/:NAME/pdus/:ID",
        "GET        /accessRights/:BASE/:NAME/serialPorts",
        "POST       /accessRights/:BASE/:NAME/serialPorts",
        "DELETE     /accessRights/:BASE/:NAME/serialPorts/:ID",
        "GET        /accessRights/:BASE/:NAME/serialPorts/:ID",
        "PATCH      /accessRights/:BASE/:NAME/serialPorts/:ID",
        "PUT        /accessRights/:BASE/:NAME/serialPorts/:ID",
        "GET        /accessRights/:BASE/:NAME/ups",
        "POST       /accessRights/:BASE/:NAME/ups",
        "DELETE     /accessRights/:BASE/:NAME/ups/:ID",
        "GET        /accessRights/:BASE/:NAME/upsOutletGroups",
        "POST       /accessRights/:BASE/:NAME/upsOutletGroups",
        "DELETE     /accessRights/:BASE/:NAME/upsOutletGroups/:ID",
        "GET        /accessRights/dsview",
        "PATCH      /accessRights/dsview",
        "PUT        /accessRights/dsview",
        "GET        /authentication",
        "PATCH      /authentication",
        "PUT        /authentication",
        "GET        /authentication/dsview",
        "PATCH      /authentication/dsview",
        "PUT        /authentication/dsview",
        "GET        /authentication/duo",
        "PATCH      /authentication/duo",
        "PUT        /authentication/duo",
        "GET        /authentication/kerberos",
        "PATCH      /authentication/kerberos",
        "PUT        /authentication/kerberos",
        "GET        /authentication/ldap",
        "PATCH      /authentication/ldap",
        "PUT        /authentication/ldap",
        "GET        /authentication/radius",
        "PATCH      /authentication/radius",
        "PUT        /authentication/radius",
        "GET        /authentication/tacacs",
        "PATCH      /authentication/tacacs",
        "PUT        /authentication/tacacs",
        "GET        /auxPorts",
        "GET        /auxPorts/:PORT",
        "PATCH      /auxPorts/:PORT",
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
"PUT        /auxPorts/:PORT",
"GET        /casProfile",
"PATCH      /casProfile",
"PUT        /casProfile",
"GET        /casProfile/autoAnswer",
"POST       /casProfile/autoAnswer",
"DELETE     /casProfile/autoAnswer/:ID",
"GET        /casProfile/autoAnswer/:ID",
"GET        /casProfile/commands",
"POST       /casProfile/commands",
"DELETE     /casProfile/commands/:NAME",
"GET        /casProfile/commands/:NAME",
"PATCH      /casProfile/commands/:NAME",
"PUT        /casProfile/commands/:NAME",
"GET        /casProfile/matchStrings",
"POST       /casProfile/matchStrings",
"DELETE     /casProfile/matchStrings/:ID",
"GET        /casProfile/matchStrings/:ID",
"GET        /casProfile/poolOfPorts",
"POST       /casProfile/poolOfPorts",
"DELETE     /casProfile/poolOfPorts/:NAME",
"GET        /casProfile/poolOfPorts/:NAME",
"PATCH      /casProfile/poolOfPorts/:NAME",
"PUT        /casProfile/poolOfPorts/:NAME",
"GET        /casProfile/probeStrings",
"POST       /casProfile/probeStrings",
"DELETE     /casProfile/probeStrings/:ID",
"GET        /casProfile/probeStrings/:ID",
"POST       /changePassword",
"GET        /dialinProfile",
"PATCH      /dialinProfile",
"PUT        /dialinProfile",
"GET        /dialinProfile/callbackUsers",
"POST       /dialinProfile/callbackUsers",
"DELETE     /dialinProfile/callbackUsers/:NAME",
"GET        /dialinProfile/callbackUsers/:NAME",
"PATCH      /dialinProfile/callbackUsers/:NAME",
"PUT        /dialinProfile/callbackUsers/:NAME",
"GET        /dialinProfile/callerId",
"POST       /dialinProfile/callerId",
"DELETE     /dialinProfile/callerId/:ID",
"GET        /dialinProfile/callerId/:ID",
"GET        /dialinProfile/chapSecrets",
"POST       /dialinProfile/chapSecrets",
"DELETE     /dialinProfile/chapSecrets/:NAME",
"GET        /dialinProfile/chapSecrets/:NAME",
"PATCH      /dialinProfile/chapSecrets/:NAME",
"PUT        /dialinProfile/chapSecrets/:NAME",
"GET        /dialinProfile/pppOtpUsers",
"POST       /dialinProfile/pppOtpUsers",
"DELETE     /dialinProfile/pppOtpUsers/:NAME",
"GET        /dialinProfile/pppOtpUsers/:NAME",
"GET        /digitalOut",
"GET        /digitalOut/:ID",
"PATCH      /digitalOut/:ID",
"PUT        /digitalOut/:ID",
"GET        /events",
"GET        /events/:ID",
"PATCH      /events/:ID",
```

```
"PUT        /events/:ID",
"GET        /events/applianceLogging",
"PATCH      /events/applianceLogging",
"PUT        /events/applianceLogging",
"GET        /events/dataBuffering",
"PATCH      /events/dataBuffering",
"PUT        /events/dataBuffering",
"GET        /events/dsview",
"PATCH      /events/dsview",
"PUT        /events/dsview",
"GET        /events/email",
"PATCH      /events/email",
"PUT        /events/email",
"GET        /events/sms",
"PATCH      /events/sms",
"PUT        /events/sms",
"GET        /events/snmp",
"PATCH      /events/snmp",
"PUT        /events/snmp",
"GET        /events/syslog",
"PATCH      /events/syslog",
"PUT        /events/syslog",
"POST       /events/test”,
"GET        /events/trapForward",
"POST       /events/trapForward",
"DELETE     /events/trapForward/:ID",
"GET        /events/trapForward/:ID",
"PATCH      /events/trapForward/:ID",
"PUT        /events/trapForward/:ID",
"GET        /events/trapForward/settings",
"PATCH      /events/trapForward/settings",
"PUT        /events/trapForward/settings",
"GET        /groups",
"POST       /groups",
"DELETE     /groups/:NAME",
"GET        /groups/:NAME",
"PATCH      /groups/:NAME",
"PUT        /groups/:NAME",
"GET        /modems",
"GET        /modems/:PORT",
"PATCH      /modems/:PORT",
"PUT        /modems/:PORT",
"POST       /modems/:PORT/registration",
"POST       /modems/:PORT/signalCheck",
"GET        /monitoring/autoDiscovery",
"GET        /monitoring/autoDiscovery/:PORT",
"POST       /monitoring/autoDiscovery/:PORT/discoverNow",
"GET        /monitoring/callerIdLog",
"POST       /monitoring/callerIdLog/clearLog",
"GET        /monitoring/fipsMode",
"GET        /monitoring/ipsec",
"GET        /monitoring/ipsec/:NAME",
"GET        /monitoring/modemPppdLog",
"POST       /monitoring/modemPppdLog/clearLog",
"GET        /monitoring/network/devices",
"GET        /monitoring/network/devices/:INT",
"GET        /monitoring/network/routingTables/:TABLE",
"GET        /monitoring/scheduledTasks",
"POST       /monitoring/scheduledTasks",
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
"DELETE      /monitoring/scheduledTasks/:NAME",
"GET         /monitoring/scheduledTasks/:NAME",
"PATCH       /monitoring/scheduledTasks/:NAME",
"PUT         /monitoring/scheduledTasks/:NAME",
"POST        /monitoring/scheduledTasks/:NAME/runNow",
"GET         /monitoring/serialPorts",
"DELETE      /monitoring/serialPorts/:ID",
"GET         /monitoring/serialPorts/:ID",
"GET         /monitoring/zeroTouchLog",
"POST        /monitoring/zeroTouchLog/clearLog",
"GET        /network/certificates/root",
"DELETE      /network/certificates/root/#NAME",
"GET         /network/certificates/root/#NAME",
"POST        /network/certificates/root/download",
"POST        /network/certificates/root/upload",
"GET         /network/certificates/settings",
"PATCH       /network/certificates/settings",
"PUT         /network/certificates/settings",
"GET         /network/certificates/status",
"GET         /network/devices",
"GET         /network/devices/:INT",
"PATCH       /network/devices/:INT",
"PUT         /network/devices/:INT",
"GET         /network/dhcpServerAddresses",
"POST        /network/dhcpServerAddresses",
"DELETE      /network/dhcpServerAddresses/#ADDR",
"GET         /network/dhcpServerAddresses/#ADDR",
"PATCH       /network/dhcpServerAddresses/#ADDR",
"PUT         /network/dhcpServerAddresses/#ADDR",
"GET         /network/dhcpServerSettings",
"PATCH       /network/dhcpServerSettings",
"PUT         /network/dhcpServerSettings",
"GET         /network/firewall/:TABLE",
"POST        /network/firewall/:TABLE",
"DELETE      /network/firewall/:TABLE/:NAME",
"GET         /network/firewall/:TABLE/:NAME",
"PATCH       /network/firewall/:TABLE/:NAME",
"PUT         /network/firewall/:TABLE/:NAME",
"GET         /network/firewall/:TABLE/:NAME/rules",
"POST        /network/firewall/:TABLE/:NAME/rules",
"DELETE      /network/firewall/:TABLE/:NAME/rules/:ID",
"GET         /network/firewall/:TABLE/:NAME/rules/:ID",
"PATCH       /network/firewall/:TABLE/:NAME/rules/:ID",
"PUT         /network/firewall/:TABLE/:NAME/rules/:ID",
"POST        /network/firewall/:TABLE/:NAME/rules/:ID/move",
"GET         /network/gre/tunnels",
"POST        /network/gre/tunnels ",
"DELETE      /network/gre/tunnels /:NAME",
"GET         /network/gre/tunnels /:NAME",
"PATCH       /network/gre/tunnels /:NAME",
"PUT         /network/gre/tunnels /:NAME",
"POST        /network/gre/tunnels /:NAME/connect",
"GET         /network/gre/tunnels /:NAME/diagnostics",
"POST        /network/gre/tunnels /:NAME/disconnect",
"POST        /network/gre/tunnels /:NAME/ping",
"GET         /network/hosts",
"POST        /network/hosts",
"DELETE      /network/hosts/#ADDR",
"GET         /network/hosts/#ADDR",
```

```
"PATCH      /network/hosts/#ADDR",
"PUT        /network/hosts/#ADDR",
"GET        /network/ipsec/certificates",
"DELETE     /network/ipsec/certificates/#NAME",
"POST       /network/ipsec/certificates/download",
"GET        /network/ipsec/connections",
"POST       /network/ipsec/connections",
"DELETE     /network/ipsec/connections/:NAME",
"GET        /network/ipsec/connections/:NAME",
"PATCH      /network/ipsec/connections/:NAME",
"PUT        /network/ipsec/connections/:NAME",
"POST       /network/ipsec/connections/:NAME/connect",
"GET        /network/ipsec/connections/:NAME/diagnostics",
"POST       /network/ipsec/connections/:NAME/disconnect",
"POST       /network/ipsec/connections/:NAME/ping",
"GET        /network/settings",
"PATCH      /network/settings",
"PUT        /network/settings",
"GET        /network/snmp",
"POST       /network/snmp",
"DELETE     /network/snmp/#ID",
"GET        /network/snmp/#ID",
"PATCH      /network/snmp/#ID",
"PUT        /network/snmp/#ID",
"GET        /network/snmp/system",
"PATCH      /network/snmp/system",
"PUT        /network/snmp/system",
"GET        /network/staticRoutes/:TABLE",
"POST       /network/staticRoutes/:TABLE",
"DELETE     /network/staticRoutes/:TABLE/*ID",
"GET        /network/staticRoutes/:TABLE/*ID",
"PATCH      /network/staticRoutes/:TABLE/*ID",
"PUT        /network/staticRoutes/:TABLE/*ID",
"GET        /pluggableDevices",
"GET        /pluggableDevices/:NAME",
"POST       /pluggableDevices/:NAME/delete",
"POST       /pluggableDevices/:NAME/eject",
"POST       /pluggableDevices/:NAME/setConsole",
"GET        /power/login",
"PATCH      /power/login",
"PUT        /power/login",
"GET        /power/networkPdus",
"POST       /power/networkPdus",
"DELETE     /power/networkPdus/#ADDR",
"GET        /power/networkPdus/#ADDR",
"PATCH      /power/networkPdus/#ADDR",
"PUT        /power/networkPdus/#ADDR",
"GET        /power/networkUps",
"POST       /power/networkUps",
"DELETE     /power/networkUps/#ADDR",
"GET        /power/networkUps/#ADDR",
"PATCH      /power/networkUps/#ADDR",
"PUT        /power/networkUps/#ADDR",
"GET        /power/outletGroups",
"POST       /power/outletGroups",
"DELETE     /power/outletGroups/:NAME",
"GET        /power/outletGroups/:NAME",
"POST       /power/outletGroups/:NAME/cycle",
"POST       /power/outletGroups/:NAME/off",
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
"POST      /power/outletGroups/:NAME/on",
"GET       /power/outletGroups/:NAME/outlets",
"POST      /power/outletGroups/:NAME/outlets",
"DELETE    /power/outletGroups/:NAME/outlets/:ID",
"GET       /power/pdus",
"GET       /power/pdus/:NAME",
"PATCH     /power/pdus/:NAME",
"PUT       /power/pdus/:NAME",
"GET       /power/pdus/:NAME/banks",
"GET       /power/pdus/:NAME/banks/:ID",
"PATCH     /power/pdus/:NAME/banks/:ID",
"PUT       /power/pdus/:NAME/banks/:ID",
"POST      /power/pdus/:NAME/banks/:ID/resetValues",
"POST      /power/pdus/:NAME/cycle",
"POST      /power/pdus/:NAME/factoryDefaults",
"POST      /power/pdus/:NAME/firmwareDownload",
"POST      /power/pdus/:NAME/firmwareInstall",
"GET       /power/pdus/:NAME/inlets",
"GET       /power/pdus/:NAME/inlets/:ID",
"POST      /power/pdus/:NAME/off",
"POST      /power/pdus/:NAME/on",
"GET       /power/pdus/:NAME/outlets",
"GET       /power/pdus/:NAME/outlets/:ID",
"PATCH     /power/pdus/:NAME/outlets/:ID",
"PUT       /power/pdus/:NAME/outlets/:ID",
"POST      /power/pdus/:NAME/outlets/:ID/cycle",
"POST      /power/pdus/:NAME/outlets/:ID/lock",
"POST      /power/pdus/:NAME/outlets/:ID/off",
"POST      /power/pdus/:NAME/outlets/:ID/on",
"POST      /power/pdus/:NAME/outlets/:ID/resetValues",
"POST      /power/pdus/:NAME/outlets/:ID/unlock",
"GET       /power/pdus/:NAME/phases",
"GET       /power/pdus/:NAME/phases/:ID",
"PATCH     /power/pdus/:NAME/phases/:ID",
"PUT       /power/pdus/:NAME/phases/:ID",
"POST      /power/pdus/:NAME/phases/:ID/resetValues",
"POST      /power/pdus/:NAME/reboot",
"POST      /power/pdus/:NAME/refresh",
"POST      /power/pdus/:NAME/rename",
"POST      /power/pdus/:NAME/resetValues",
"GET       /power/pdus/:NAME/sensors",
"GET       /power/pdus/:NAME/sensors/:ID",
"PATCH     /power/pdus/:NAME/sensors/:ID",
"PUT       /power/pdus/:NAME/sensors/:ID",
"POST      /power/pdus/:NAME/sensors/:ID/resetValues",
"GET       /power/ups",
"GET       /power/ups/:NAME",
"PATCH     /power/ups/:NAME",
"PUT       /power/ups/:NAME",
"GET       /power/ups/:NAME/batteryModules",
"GET       /power/ups/:NAME/batteryModules/:ID",
"GET       /power/ups/:NAME/outletGroups",
"GET       /power/ups/:NAME/outletGroups/:ID",
"POST      /power/ups/:NAME/outletGroups/:ID/cycle",
"POST      /power/ups/:NAME/outletGroups/:ID/off",
"POST      /power/ups/:NAME/outletGroups/:ID/on",
"POST      /power/ups/:NAME/outputCycle",
"POST      /power/ups/:NAME/outputOff",
"POST      /power/ups/:NAME/outputOn",
```

```
"GET        /power/ups/:NAME/powerModules",
"GET        /power/ups/:NAME/powerModules/:ID",
"POST       /power/ups/:NAME/rename",
"POST       /power/ups/:NAME/resetPowerStats",
"POST       /power/ups/:NAME/restoreName",
"POST       /power/ups/:NAME/silenceAlarm",
"POST       /power/ups/:NAME/testBattery",
"GET        /resources",
"GET        /security",
"PATCH      /security",
"PUT        /security",
"POST       /security/clearDSView",
"GET        /sensors/1Wire",
"DELETE     /sensors/1Wire/:ADDR",
"GET        /sensors/1Wire/:ADDR",
"PATCH      /sensors/1Wire/:ADDR",
"PUT        /sensors/1Wire/:ADDR",
"POST       /sensors/1Wire/refresh",
"GET        /sensors/digitalIn",
"GET        /sensors/digitalIn/:ID",
"PATCH      /sensors/digitalIn/:ID",
"PUT        /sensors/digitalIn/:ID",
"GET        /sensors/internal",
"PATCH      /sensors/internal",
"PUT        /sensors/internal",
"GET        /sensors/pdu",
"POST       /sensors/pdu/#NAME/reset",
"GET        /serialPorts",
"GET        /serialPorts/:PORT",
"PATCH      /serialPorts/:PORT",
"PUT        /serialPorts/:PORT",
"GET        /serialPorts/:PORT/alerts",
"POST       /serialPorts/:PORT/alerts",
"DELETE     /serialPorts/:PORT/alerts/:ID",
"POST       /serialPorts/:PORT/alerts/clear",
"POST       /serialPorts/:PORT/alerts/deleteAny",
"POST       /serialPorts/:PORT/clone",
"POST       /serialPorts/:PORT/disable",
"POST       /serialPorts/:PORT/discoverNow",
"POST       /serialPorts/:PORT/enable",
"GET        /serialPorts/:PORT/power",
"POST       /serialPorts/:PORT/power",
"DELETE     /serialPorts/:PORT/power/:ID",
"POST       /serialPorts/:PORT/resetToFactory",
"GET        /sessions",
"DELETE     /sessions/:ID",
"GET        /sessions/:ID",
"POST       /sessions/login",
"POST       /sessions/logout",
"GET        /sessions/refresh",
"GET        /system/bootConfig",
"PATCH      /system/bootConfig",
"PUT        /system/bootConfig",
"GET        /system/certificate",
"POST       /system/certificate/apply",
"POST       /system/certificate/download",
"POST       /system/certificate/generate",
"POST       /system/certificate/upload",
"POST       /system/config/restore",
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

```
        "POST        /system/config/save",
        "GET         /system/dateAndTime",
        "PATCH       /system/dateAndTime",
        "PUT         /system/dateAndTime",
        "POST        /system/dateAndTime/testNtp",
        "GET         /system/dateAndTime/timezone/custom",
        "PATCH       /system/dateAndTime/timezone/custom",
        "PUT         /system/dateAndTime/timezone/custom",
        "GET         /system/dateAndTime/timezones",
        "POST        /system/factoryDefault",
        "GET         /system/firmware",
        "POST        /system/firmware/download",
        "GET         /system/firmware/downloaded",
        "POST        /system/firmware/install",
        "GET         /system/firmware/version",
        "GET         /system/general",
        "PATCH       /system/general",
        "PUT         /system/general",
        "GET         /system/info",
        "POST        /system/integrity/generate",
        "POST        /system/integrity/verify",
        "POST        /system/reboot",
        "POST        /system/shutdown",
        "GET         /system/usage/flash",
        "GET         /system/usage/memory",
        "GET         /users",
        "POST        /users",
        "DELETE      /users/:NAME",
        "GET         /users/:NAME",
        "PATCH       /users/:NAME",
        "PUT         /users/:NAME",
        "POST        /users/:NAME/disable",
        "POST        /users/:NAME/enable",
        "POST        /users/:NAME/unlock",
        "GET         /users/passwordRules",
        "PATCH       /users/passwordRules",
        "PUT         /users/passwordRules"
    ]
}
```

Appendices

Appendix A: cURL

The cURL command line utility is one method that can be used to communicate with the RESTful API.

Example using JWT authentication

```
$ curl -H "Content-Type: application/json" -H "Accept:application/json"
https://10.20.30.40:48048/api/v1/sessions/login -d
 '{"username":"admin","password":"avocent"}'

{
  "token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJleHAiOjE1MDQxMTU2NzUsImlkIjoiYWRtaW4iLCJvcmlnX2lhdC
I6MTUwNDExMjA3NSwic2lkIjo4fQ.UYGXje5It2hAJryruP3etUaabSh5pfiPP_sXXZF37og"
}

$ curl -H "Content-Type: application/json" -H "Accept:application/json" -H
"Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJleHAiOjE1MDQxMTU2NzUsImlkIjoiYWRta
W4iLCJvcmlnX2lhdCI6MTUwNDExMjA3NSwic2lkIjo4fQ.UYGXje5It2hAJryruP3etUaabSh5
pfiPP_sXXZF37og" https://10.20.30.40:48048/api/v1/system/info
{
  "serialNumber": "0012345678",
  "type": "ACS8048 with single power supply",
  "bootcode": "1.17",
  "firmware": "1.3.77.2909+551+28+11",
  "bootedFrom": "hardware",
  "powerSupply1": "on",
  "cpu": "ARMv7 Processor rev 0 (v7l)",
  "cores": 2
}
```

Examples using basic authentication

```
$ curl -k –basic –user admin:avocent
https://10.20.30.40:48048/api/v1/system/info?fields=serialNumber

{

      "serialNumber": "0530030233"

}


$ curl -k --basic --user admin:avocent -H "Content-Type: application/json"
 -X PUT https://10.20.30.40:48048/api/v1/security -d
'{"idleTimeout":1234, "oneWire":"disabled"}'
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

Appendix B: Python

Python examples are included on the console system in the /usr/local/examples directory for both HTTP and HTTPS that demonstrate how python can be used to communicate with the RESTful API. These examples run on Python versions 2.7 and 3.5. They require the python "requests" module be installed on the system running Python.

Example

```
$ python acsapi_example_http.py
```


Proprietary and Confidential ©2025 Vertiv Group Corp.

Appendix C: Helper Script

A bash shell helper script is provided in the Avocent ACS800/8000 advanced console system root filesystem under the /usr/share/restapi/restapi-helper.sh path. This helper script modifies the environment of the running shell to add GET/PUT/POST/PATCH/DELETE shell functions. These functions provide a simple command line interface to demonstrate the API. The shell functions utilize the cURL program, which must be installed on the system.

From a bash shell, the script must be "sourced" in order to make the necessary changes to the shell environment, which are shown as SETUP parameters and functions. When sourcing the script, the last parameter is the IPv4 address of the console system.

In the following example, 10.20.30.40 is the Avocent ACS800/8000 advanced console system IP address.

Example

```
$ . /usr/share/restapi/restapi-helper.sh 10.20.30.40
Token for Basic Authentication saved in /home/root/.acsrestapi/Basic-token-127.0.0.1
SETUP parameters:

ACSHOST          127.0.0.1
ACSPROTOCOL      http
ACSPORT          8180
ACSTOKENDIR      /home/root/.acsrestapi
ACSTOKENTYP      Basic   (Basic Authentication)
ACSURL           /api/v1
ACSDEBUG         no

Change your ACSHOST parameter to the console system's IP address.  For example:
ACSHOST=192.168.161.10

GET /resource
                Example:
                    GET /serialPorts/2

PUT /resource  '{...json parameters...}'
                Example:
                    PUT /serialPorts/2  '{"physical":{"speed":38400}}'

PATCH /resource  '{...json parameters...}'
                Example:
                    PATCH /security '{"idleTimeout":0}'

POST /resource  '{...json parameters...}'
                Example:
                    POST /sessions/login  '
{"username":"admin","password":"avocent"}'

DELETE /resource

To show a list of available 'resource' items, use:

    GET /resources
```

NOTE: These functions automatically prepend '/acs/v1' to the /resource.

Proprietary and Confidential ©2025 Vertiv Group Corp.

After running the helper script, the current shell environment can execute POST, GET and the other commands previously mentioned.

Example

```
$ . restapi-helper.sh 10.20.30.40

$ POST /sessions/login '{"username":"admin","password":"avocent"}'
/sessions/login successful. JWT saved in /home/acs/.acsrestapi/Bearer-token-10.20.30.40

$ GET /system/info
{
  "serialNumber": "001234567",
  "type": "ACS8048 with single power supply",
  "bootcode": "1.18",
  "firmware": "1.3.77.2909+551+28+11",
  "firmwareDate": "Sep 20 2017 - 08:03:39",
  "bootedFrom": "hardware",
  "powerSupply1": "on",
  "cpu": "ARMv7 Processor rev 0 (v7l)",
  "cores": 2
}
```

 Proprietary and Confidential ©2025 Vertiv Group Corp.

Appendix D: Certificate Verification

The client may wish to avoid certificate verification warnings when using HTTPS. With the Python requests package, this is done by adding verify=False to the requests command.

To disable certitificate verification with Python:

```
>>> requests.get(URL, verify=False)
```

To disable certificate verification with cURL:

```
$ curl -k -H "Content-Type: application/json" -H "Accept:application/json"
https://10.20.30.40:48048/api/v1/sessions/login -d
'{"username":"admin","password":"avocent"}'
```

The restapi-helper script automatically includes the -k option on the underlying curl commands when using HTTPS.

Proprietary and Confidential ©2025 Vertiv Group Corp.

This page intentionally left blank

Proprietary and Confidential ©2025 Vertiv Group Corp.

Connect with Vertiv on Social Media

https://www.facebook.com/vertiv/

https://www.instagram.com/vertiv/

https://www.linkedin.com/company/vertiv/

https://www.x.com/Vertiv/

Vertiv.com | Vertiv Headquarters, 505 N Cleveland Ave, Westerville, OH 43082 USA

©2025 Vertiv Group Corp. All rights reserved. Vertiv™ and the Vertiv logo are trademarks or registered trademarks of Vertiv Group Corp. All other names and logos referred to are trade names, trademarks or registered trademarks of their respective owners. While every precaution has been taken to ensure accuracy and completeness here, Vertiv Group Corp. assumes no responsibility, and disclaims all liability, for damages resulting from use of this information or for any errors or omissions.

590-1814-501G