

Developer's Guide

AudioCodes Gateway, SBC and MSBR Series

REST API for Devices

Version 7.4



Table of Contents

1	Overview	3
2	User Privilege Levels and REST API Access	5
3	Authentication and Session Establishment.....	9
4	Top-Level Folder	11
5	Navigation Tree	13
6	Actions	15
6.1	Reset Device	16
6.2	Save Configuration	17
6.3	Auth Token	17
7	Files	19
7.1	File Encoding.....	20
7.1.1	File Upload Encoding.....	20
7.1.2	File Download Encoding	22
7.2	INI File.....	22
7.2.1	Full INI File.....	22
7.2.2	Incremental INI File.....	24
7.3	CLI Script	25
7.3.1	Full CLI Script	25
7.3.2	Incremental CLI Script	27
7.4	Software Load	29
7.4.1	Hitless Software Upgrade	30
7.5	Additional Files	31
8	TLS Context Files	33
8.1	Selecting TLS Context.....	34
8.2	Private Key	35
8.2.1	Generate New Private Key	36
8.3	Device Certificate	37
8.4	Generate Self-Signed Certificate	39
8.5	Generate Certificate Signing Request.....	40
8.6	Trusted Root Certificates	41
8.7	Add Certificate to Trusted Root Store	43
9	Alarms	45
9.1	Active Alarms	46
9.1.1	Specific Active Alarm	48
9.1.2	Time of the Last Active Alarm.....	49
9.2	Alarm History.....	50
9.2.1	Specific History Alarm.....	52
10	Device Status.....	53
11	Performance Monitoring.....	55
11.1	Hierarchical Tree Structure of KPI	56
11.2	Cursor-based Pagination	56

11.3	Historical Intervals Discovery.....	59
11.4	Application, Group and Entity Discovery.....	60
11.5	Entity Index (ID) Discovery	63
11.6	Discovery of Performance Monitoring Entity	64
11.7	Specific Performance Monitoring Parameters.....	67
12	License Management.....	69
13	Full List of Supported HTTP Responses.....	73

Notice

Information contained in this document is believed to be accurate and reliable at the time of printing. However, due to ongoing product improvements and revisions, AudioCodes cannot guarantee accuracy of printed material after the Date Published nor can it accept responsibility for errors or omissions. Updates to this document can be downloaded from <https://www.audiocodes.com/library/technical-documents>.

This document is subject to change without notice.

Date Published: August-10-2021

WEEE EU Directive

Pursuant to the WEEE EU Directive, electronic and electrical waste must not be disposed of with unsorted waste. Please contact your local recycling authority for disposal of this product.

Customer Support

Customer technical support and services are provided by AudioCodes or by an authorized AudioCodes Service Partner. For more information on how to buy technical support for AudioCodes products and for contact information, please visit our website at <https://www.audiocodes.com/services-support/maintenance-and-support>.

Abbreviations and Terminology

Each abbreviation, unless widely used, is spelled out in full when first used. Device refers to SBC and gateway.

Document Revision Record

LTRT	Description
41766	Initial document release for Ver. 7.4.
41768	ConfigurationPackage

Documentation Feedback

AudioCodes continually strives to produce high quality documentation. If you have any comments (suggestions or errors) regarding this document, please fill out the Documentation Feedback form on our website at <https://online.audiocodes.com/documentation-feedback>.

1 Overview

The REST API is designed for developers who wish to programmatically integrate the Mediant Gateway or SBC device into their solution and for administrators who wish to perform management and configuration tasks via automation scripts.

The REST API provides access to the resources via pre-defined URL paths. Each resource represents specific device configuration element, state object or maintenance action.

The REST API uses standard HTTP/1.1 protocol. For enhanced security it is recommended to secure the traffic via the use of HTTPS transport layer.

Standard HTTP methods – GET, PUT, POST and DELETE – are used to read the resource's state and to create/update/delete the resources (wherever applicable). Resource state is described in JSON format and included in the HTTP request or response bodies.

This page is intentionally left blank.

2 User Privilege Levels and REST API Access

Each API URL resource (e.g., alarms/active) and HTTP method (GET, PUT, POST or DELETE) has a minimum user privilege (access) level. For example, only REST users with Security Administrator level can replace (PUT) the device's License Key.

REST users and their access levels (Monitor, Administrator, and Security Administrator) are configured in the Local Users table (like for other management interfaces).

REST users accessing through LDAP or RADIUS must have a minimum access level of 50 (read-only). For prohibited user access, the device responds with a 403 Forbidden Status.

User access to the REST API directories also depends on the user's access level:

Table 2-1: Minimum User Access Level per Directory

Directory	Minimum User Level
/alarms	Monitor
/kpi	Monitor
/status	Monitor
/actions	Administrator
/files	Administrator
/license	Administrator

For a supported HTTP method, if access is denied due to a user's access level, a 403 Forbidden Status or 405 Method Not Allowed response is sent by the device. For requested resources that do not have any content, a 400 Bad Request response is sent.

The following table lists the REST API resources and the corresponding user access level per HTTP method supported for that resource.

Table 2-2: Minimum User Access Level per REST API Resource

REST-API	HTTP Method			
	GET	PUT	POST	DELETE
/api/v1/versions	Monitor	405 Method Not Allowed	405 Method Not Allowed	405 Method Not Allowed
actions/reset	405 Method Not Allowed	405 Method Not Allowed	Administrator	405 Method Not Allowed
actions/saveConfiguration	405 Method Not Allowed	405 Method Not Allowed	Administrator	405 Method Not Allowed
actions/authToken	405 Method Not Allowed	405 Method Not Allowed	Security Administrator	405 Method Not Allowed
actions	Administrator	405 Method Not Allowed	405 Method Not Allowed	405 Method Not Allowed
status	Monitor	405 Method Not Allowed	405 Method Not Allowed	405 Method Not Allowed
license	Administrator	Security Administrator	405 Method Not Allowed	405 Method Not Allowed

REST-API	HTTP Method			
	GET	PUT	POST	DELETE
alarms/active	Monitor	405 Method Not Allowed	405 Method Not Allowed	405 Method Not Allowed
alarms/history	Monitor	405 Method Not Allowed	405 Method Not Allowed	405 Method Not Allowed
alarms	Monitor	405 Method Not Allowed	405 Method Not Allowed	405 Method Not Allowed
kpi	Monitor	405 Method Not Allowed	405 Method Not Allowed	405 Method Not Allowed
mc_status	Monitor	405 Method Not Allowed	405 Method Not Allowed	405 Method Not Allowed
files/ini	Security Administrator	Security Administrator	405 Method Not Allowed	405 Method Not Allowed
files/ini/incremental	405 Method Not Allowed	Security Administrator	405 Method Not Allowed	405 Method Not Allowed
files/cliScript/incremental	405 Method Not Allowed	Security Administrator	405 Method Not Allowed	405 Method Not Allowed
files/cliScript	Security Administrator	Security Administrator	405 Method Not Allowed	405 Method Not Allowed
files/software	405 Method Not Allowed	Administrator	405 Method Not Allowed	405 Method Not Allowed
files/software/hitless	405 Method Not Allowed	Administrator	405 Method Not Allowed	405 Method Not Allowed
files/cpt	405 Method Not Allowed	Administrator	405 Method Not Allowed	405 Method Not Allowed
files/prt	405 Method Not Allowed	Administrator	405 Method Not Allowed	405 Method Not Allowed
files/dialplan	405 Method Not Allowed	Administrator	405 Method Not Allowed	405 Method Not Allowed
files/casTable	405 Method Not Allowed	Administrator	405 Method Not Allowed	405 Method Not Allowed
files/amd	405 Method Not Allowed	Administrator	405 Method Not Allowed	405 Method Not Allowed
files/usersInfo	405 Method Not Allowed	Administrator	405 Method Not Allowed	405 Method Not Allowed
files/configurationPackage.tar.gz	Security Administrator	Security Administrator	405 Method Not Allowed	405 Method Not Allowed
files/sbcWizard	405 Method Not Allowed	Administrator	405 Method Not Allowed	405 Method Not Allowed
files/fixs	405 Method Not Allowed	Administrator	405 Method Not Allowed	405 Method Not Allowed
files/fixo	405 Method Not Allowed	Administrator	405 Method Not Allowed	405 Method Not Allowed

REST-API	HTTP Method			
	GET	PUT	POST	DELETE
files	Administrator	405 Method Not Allowed	405 Method Not Allowed	405 Method Not Allowed
files/tls	Security Administrator	Security Administrator	Security Administrator	405 Method Not Allowed

This page is intentionally left blank.

3 Authentication and Session Establishment

The REST API is accessible via HTTP/HTTPS protocol at `/api/v1` prefix.

Example

```
GET /api/v1/status HTTP/1.1
Host: 10.4.219.229
```

```
HTTP/1.1 200 OK
Content-Type: application/json
{
  "localTimeStamp": "2010-01-17T17:29:15.000Z",
  "ipAddress": "10.4.219.229",
  "subnetMask": "255.255.0.0",
  "defaultGateway": "10.4.0.1",
  "productType": "Mediant SW",
  "versionID": "7.20A.200.014",
  "protocolType": "SIP",
  "operationalState": "UNLOCKED",
  "highAvailability": "Not Operational",
  "serialNumber": "101780235059663",
  "macAddress": "fa163e6e7e1d",
  "systemUpTime": 161446
}
```

Each REST request must be authenticated using HTTP Basic Authentication. Provided credentials (username-password) must correspond to a valid device user. Availability of specific REST API endpoints depends on user privilege level. For more information on REST API and user privilege levels, see Section User Privilege Levels and REST API Access on page 5.



Note: It is strongly recommended to use the HTTPS transport layer when accessing the REST API to mitigate security risks.

This page is intentionally left blank.

4 Top-Level Folder

The `/api` URL serves as a root folder for accessing the REST API.

URL

`/api`

HTTP Method

`GET`

HTTP Response

`200 OK`

Example

```
GET /api HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "versions": [
    {
      "id": "v1",
      "status": "stable",
      "url": "/api/v1"
    }
  ]
}
```

This page is intentionally left blank.

5 Navigation Tree

The `/api/v1` URL displays the complete navigation tree that is supported by the REST API. This tree is displayed below:

```

/api/v1
  /actions
    /reset // reset the device
    /saveConfiguration // save configuration to NVRAM
    /authToken // get authentication token
  /files // files upload/download
    /ini // INI configuration file
    /incremental // incremental INI file
    /software // CMP software file
    /cliScript // CLI script
    /incremental // incremental CLI script
    /tls/<id> // TLS context files
    /privateKey // private key
    /certificate // device certificate
    /request // certificate signing request
    /trustedRoot // trusted root
    /... // other (auxiliary) files
  /alarms
    /active // active alarms
    /history // history alarms
  /license // license management
  /kpi // performance monitoring
    /current // current PMs
    /history // history PMs
    /interval // interval of history PM
  /status // device status

```

URL

```
/api/v1
```

HTTP Method

```
GET
```

HTTP Response

```
200 OK
```

Example

```

GET /api/v1 HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{

```

```
"items": [  
  {  
    "id": "actions",  
    "description": "Device actions",  
    "url": "/api/v1/actions"  
  },  
  {  
    "id": "alarms",  
    "description": "Device alarms",  
    "url": "/api/v1/actions"  
  },  
  {  
    "id": "files",  
    "description": "Upload and download of  
configuration files",  
    "url": "/api/v1/files"  
  },  
  {  
    "id": "license",  
    "description": "License management",  
    "url": "/api/v1/license"  
  },  
  {  
    "id": "status",  
    "description": "Device status",  
    "url": "/api/v1/status"  
  },  
  {  
    "id": "kpi",  
    "description": "key indicators performance",  
    "url": "/api/v1/kpi"  
  }  
]  
}
```

6 Actions

The `/actions` URL provides the ability to perform maintenance actions on the device.

URL

```
/api/v1/actions
```

HTTP Method

```
GET
```

HTTP Response

```
200 OK
```

Example

```
GET /api/v1/actions HTTP/1.1
```

```
Host: 10.4.219.229
```

```
HTTP/1.1 200 OK
```

```
Content-Type: application/json
```

```
{
  "actions": [
    {
      "id": "reset",
      "description": "Reset device",
      "url": "/api/v1/actions/reset"
    },
    {
      "id": "saveConfiguration",
      "description": "Save device configuration to
NVRAM",
      "url": "/api/v1/actions/saveConfiguration"
    },
    {
      "id": "authToken",
      "description": "Get authentication token",
      "url": "/api/v1/actions/authToken"
    }
  ]
}
```

6.1 Reset Device

The `/actions/reset` URL performs a device reset.

URL

`/api/v1/actions/reset`

HTTP Method

POST

Supported Request JSON Attributes

Attribute	Type	Value	Description
saveConfiguration	Boolean	true	Store current configuration before reset (default).
		false	Do not store current configuration.
gracefulTimeout	Number	0	Perform reset immediately (default).
		1	Wait for all calls to finish, and then perform reset.
		<sec>	Wait for specified time (in seconds) for calls to finish, and then perform reset.

HTTP Responses

- 200 OK
- 400 Bad request – provided attributes or values are incorrect.
- 409 Conflict – reset can't be performed due to current device state (e.g. synchronization with the redundant device is in progress).

Example

```

POST /api/v1/actions/reset HTTP/1.1
Host: 10.4.219.229
Content-Type: application/json
{
  "saveConfiguration": true,
  "gracefulTimeout": 0
}

HTTP/1.1 200 OK
Content-Type: application/json
{
  "description": "Device will reset now"
}

or

HTTP/1.1 409 Conflict
    
```

```
Content-Type: application/json
{
  "description": "Device is currently performing HA
synchronization"
}
```

6.2 Save Configuration

The `/actions/saveConfiguration` URL saves the device configuration to the non-volatile memory so that it'll be preserved if the device reboots or is powered down.

URL

```
/api/v1/actions/saveConfiguration
```

HTTP Method

```
POST
```

HTTP Responses

- 200 OK
- 409 Conflict – configuration can't be save due to current device state.

Example

```
POST /api/v1/actions/saveConfiguration HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
```

6.3 Auth Token

The `/actions/authToken` URL enables the retrieval of an authentication token that may be used to access device's Web interface without need to enter a username and a password. The generated authentication token has a limited lifetime and should be used within ten seconds after generation. In order to use the token, append it to device's URL as `authToken` parameter:

<http://10.3.4.10/index.html?mode=web&authToken=4675cd93ab9f80f45a4ec0a934f81097>

URL

```
/api/v1/actions/authToken
```

HTTP Method

```
POST
```

Supported Request JSON Attributes

Attribute	Type	Value	Description
username	String		Username for new session – to be used for activity logging and graphical display.
privLevel	String	admin operator monitor	Privilege level for new session. <ul style="list-style-type: none"> admin – security administrator operator – operator with administrative privileges (can alter configuration) monitor – monitor user (can only view configuration)
sessionTimeout	Integer		Session timeout in seconds. (optional)
crossHost	String		IP address or hostname of 3 rd party Web interface that integrates device’s Web interface via IFRAME directive. Needed to prevent cross-site request forgery (CSRF) attacks. (optional)

HTTP Response

200 OK

Example

```

POST /api/v1/actions/authToken HTTP/1.1
Host: 10.4.219.229
Content-Type: application/json
{
  "username": "john",
  "privLevel": "admin",
  "sessionTimeout": 180,
  "crossHost": "10.3.2.40"
}

HTTP/1.1 200 OK
Content-Type: application/json
{
  "authToken": "4675cd93ab9f80f45a4ec0a934f81097",
  "description": "Authentication token successfully
generated"
}
    
```

7 Files

The `/files` URL provides access to the various device configuration files.

The PUT method is used to modify the specific configuration file.

The GET method is used to get the specific configuration file (for files which support it).

URL

```
/api/v1/files
```

HTTP Method

```
GET
```

HTTP Response

```
200 OK
```

Example

```
GET /api/v1/files HTTP/1.1
```

```
Host: 10.4.219.229
```

```
HTTP/1.1 200 OK
```

```
Content-Type: application/octet-stream
```

```
{
  "files": [
    {
      "id": "ini",
      "description": "INI configuration file",
      "url": "/api/v1/files/ini"
    },
    {
      "id": "software",
      "description": "Software load",
      "url": "/api/v1/files/software"
    },
    {
      "id": "cliScript",
      "description": "CLI configuration script",
      "url": "/api/v1/files/cliScript"
    },
    ...
  ]
}
```

7.1 File Encoding

7.1.1 File Upload Encoding

File upload (PUT) operations use multipart/form-data encoding.

Example

```
PUT /api/v1/files/cliScript/incremental HTTP/1.1
Host: 10.4.219.229
Authorization: Basic QWRtaW46QWRtaW4=
Content-Length: 210
Content-Type: multipart/form-data; boundary=----
WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file"; filename="cli.txt"
Content-Type: application/octet-stream

show system version

-----WebKitFormBoundary7MA4YWxkTrZu0gW--
```

Use the following code snippets to generate proper format.

cURL

```
curl -i -X PUT -F "file=@cli.txt" -H "Expect:" --user Admin:Admin \
http://10.4.219.229/api/v1/files/cliScript/incremental
```

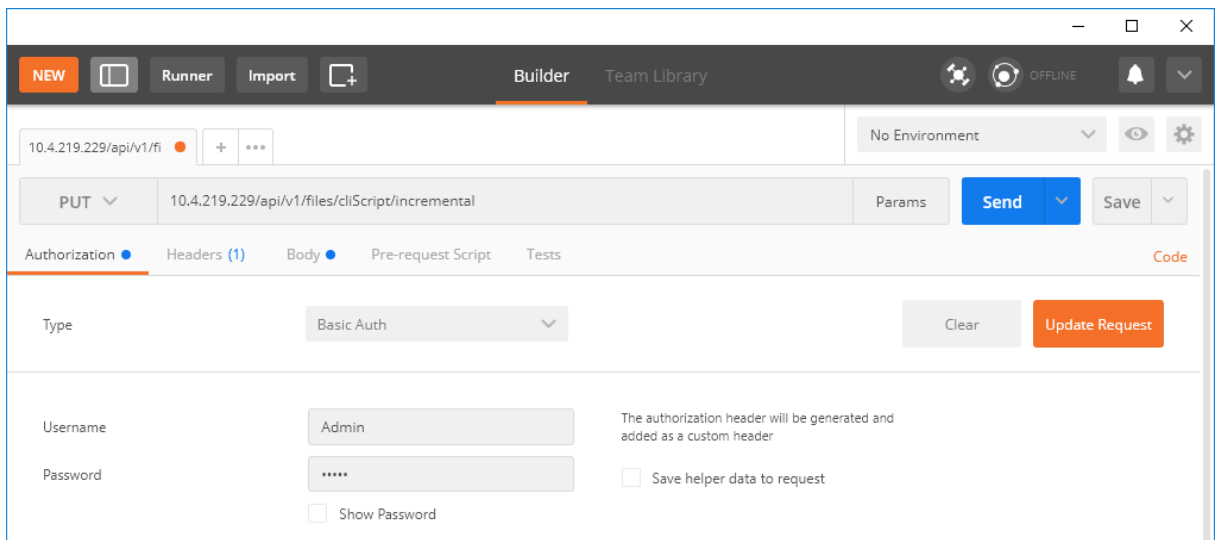
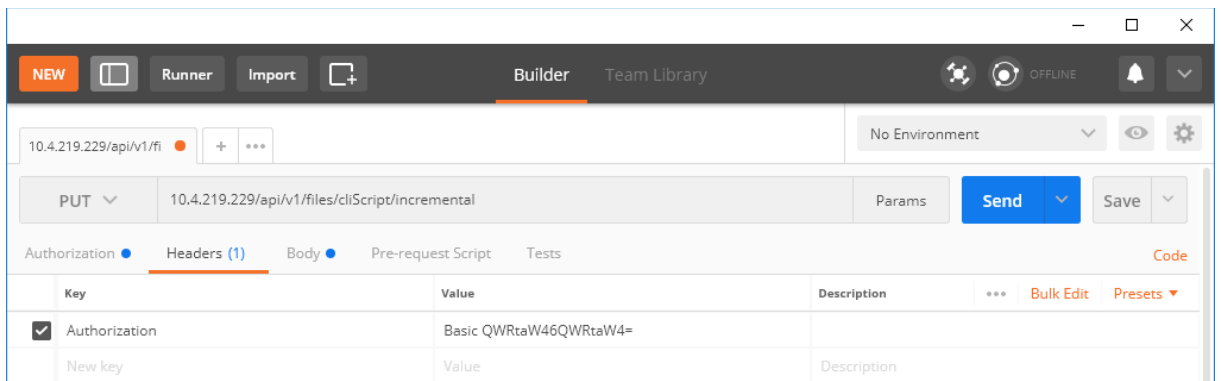
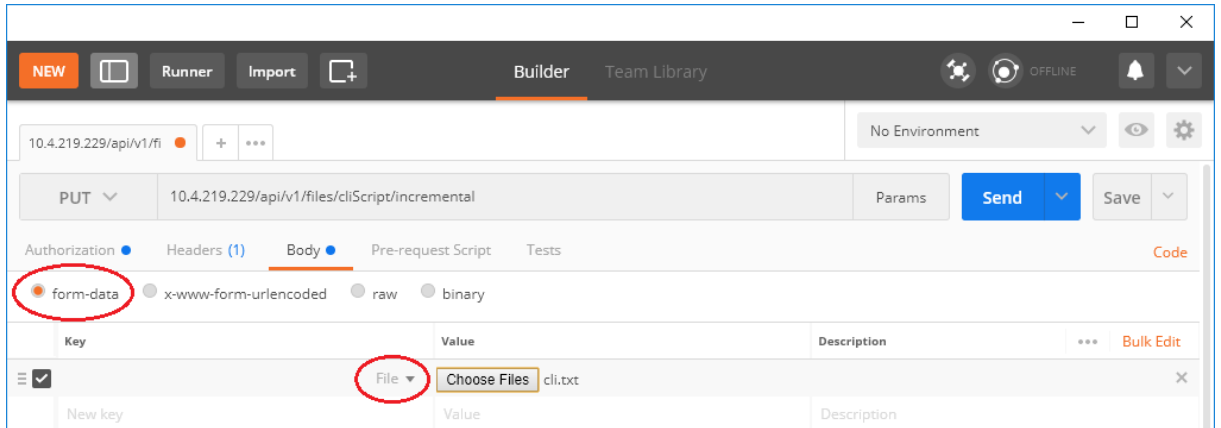
Python

```
import requests
import base64

def send_cli(ip, username, password, cli_script):
    url = 'http://' + ip + '/api/v1/files/cliScript/incremental'
    headers = {'Authorization': base64.b64encode(username + ':' +
password)}
    files = {'file': ('cli.txt', cli_script)}
    response = requests.put(url, files=files, headers=headers)
    return response.text

send_cli('10.4.219.229', 'Admin', 'Admin', 'show system version')
```


If you prefer to use the GUI tool, use Postman (<https://www.getpostman.com>) application or Chrome extension and set it up as follows:



7.1.2 File Download Encoding

Download (GET) operations use application/octet-stream encoding.

Example

```

GET /api/v1/files/ini HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/octet-stream

;*****
;** Ini File **
;*****

;Board: Mediant SW
;Board Type: 73
;Serial Number: 101780235059663
;Slot Number: 1
;Software Version: 7.20A.200.014
;DSP Software Version: SOFTDSP => 0.00
;Board IP Address: 10.4.219.229
;Board Subnet Mask: 255.255.0.0
...

```

7.2 INI File

The INI file is the main device configuration file.

7.2.1 Full INI File

The `/files/ini` URL provides the ability to upload or download an ini configuration file. Uploading of an ini file triggers device reset to activate the new configuration. Use `/files/ini/incremental` (see Section 7.2.2) to apply a partial configuration that doesn't require device reset.

URL

```
/api/v1/files/ini
```

HTTP Method

```
GET, PUT
```

HTTP Responses

- 200 OK
- 400 Bad request - provided ini file is incorrect.
- 409 Conflict – ini file can't be loaded due to the current device state (e.g. synchronization with the redundant device is in progress).

Example

```
GET /api/v1/files/ini HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/octet-stream
<INI file>
```

Example

```
PUT /api/v1/files/ini HTTP/1.1
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=----
WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file"; filename="ini.txt"
Content-Type: application/octet-stream

<INI File>
-----WebKitFormBoundary7MA4YWxkTrZu0gW--

HTTP/1.1 200 OK
Content-Type: application/json
{
  "description": "Device will reset now to activate new
configuration"
}

or

HTTP/1.1 409 Conflict
Content-Type: application/json
{
  "description": "Device is currently performing HA
synchronization"
}
```



Note: The uploaded file gets transformed by the device; therefore the file content will differ when you download it.

7.2.2 Incremental INI File

The `/files/ini/incremental` URL provides the ability to upload an incremental (partial) ini file that can be applied to the device without reset.

URL

```
/api/v1/files/ini/incremental
```

HTTP Method

```
PUT
```

HTTP Responses

- 200 OK
- 400 Bad request - provided ini file is incorrect.
- 409 Conflict – ini file can't be loaded due to the current device state (e.g. synchronization with the redundant device is in progress).

Example

```
PUT /api/v1/files/ini/incremental HTTP/1.1
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=-----WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file"; filename="ini.txt"
Content-Type: application/octet-stream

<INI File>
-----WebKitFormBoundary7MA4YWxkTrZu0gW--

HTTP/1.1 200 OK
```

7.3 CLI Script

The CLI configuration script is an alternative method (to the ini file) for detailing the device configuration.

7.3.1 Full CLI Script

The `/files/cliScript` URL provides the ability to upload or download a CLI configuration script. Uploading of a CLI script triggers device reset to activate the new configuration. Use `/files/cliScript/incremental` (see Section 7.2.2) to apply a partial configuration that doesn't require device reset.



Note: The full CLI script completely overrides the current device configuration in the same manner as the “copy startup-script from” CLI command. The provided script must contain configuration commands only and is typically generated by “show running-config” CLI command. If you need to run “show” commands or update device configuration – use incremental CLI script instead as described in Section 7.3.2.

URL

```
/api/v1/files/cliScript
```

HTTP Methods

```
GET, PUT
```

HTTP Responses

- 200 OK
- 400 Bad request - provided CLI script is incorrect.
- 409 Conflict – CLI script can't be loaded due to the current device state (e.g. synchronization with the redundant device is in progress).

Example

```
GET /api/v1/files/cliScript HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/octet-stream
<CLI script>
```

Example

```
PUT /api/v1/files/ini/incremental HTTP/1.1
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=----
WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
```

```
Content-Disposition: form-data; name="file"; filename="cli.txt"
Content-Type: application/octet-stream

<INI File>
-----WebKitFormBoundary7MA4YWxkTrZu0gW--

HTTP/1.1 200 OK
Content-Type: application/json
{
  "description": "Device will reset now to activate new
configuration"
}

or

HTTP/1.1 409 Conflict
Content-Type: application/json
{
  "description": "Device is currently performing HA
synchronization"
}
```



Note: The uploaded file gets transformed by the device; therefore the file content will differ when you download it.

7.3.2 Incremental CLI Script

The `/files/cliScript/incremental` URL provides the ability to upload an incremental (partial) CLI script that can be applied to the device without reset.

The script may contain both configuration and “show” commands. Output of the script will be returned in the response.



Note: The incremental CLI script may not contain “action” commands that require user interaction and/or take long time. For example, the “copy” command is not supported in CLI script passed via REST API. If you need to trigger file transfer initiated by the device, use Automatic Update configuration instead – e.g.:

```
configure system
  automatic-update
    firmware http://audc.com/ssbc_7.20A.200.014.cmp
```

URL

```
/api/v1/files/cliScript/incremental
```

HTTP Method

```
PUT
```

Request Content-Type

```
application/json
```

HTTP Responses

- 200 OK
- 400 Bad request – provided CLI script is incorrect.
- 409 Conflict – CLI script can’t be loaded due to current device state (e.g. synchronization with the redundant device is in progress).

Example

```
PUT /api/v1/files/cliScript/incremental HTTP/1.1
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file"; filename="cli.txt"
Content-Type: application/octet-stream

show system version
-----WebKitFormBoundary7MA4YWxkTrZu0gW--

HTTP/1.1 200 OK
Content-Type: application/json
```

```
{
  "description": "Incremental CLI Script file was loaded.",
  "output": ". Version info:. -----. ;Board:
Mediant SW. . ;Board Type: 73. . ;Serial Number:
137172915378947. . ;Slot Number: 1. . ;Software Version:
7.20A.201.357. . ;ISO Version: Mediant Software E-SBC (ver
7.20A.156.028). . ;DSP Software Version: SOFTDSP => 0.00. .
;Board IP Address: 10.4.219.242. . ;Board Subnet Mask:
255.255.0.0. . ;Board Default Gateway: 10.4.0.1. . ;Ram size:
3829M Flash size: 0M. . ;Num of DSP Cores: 1 Num DSP
Channels: 1022. . ;Profile: NONE . . ;;;Key features;;Board
Type: Mediant SW ;Max SW Ver: 9.80;FXSPorts=0 ;FXOPorts=0 ;QOE
features: VoiceQualityMonitoring MediaEnhancement ;DATA
features: ;Channel Type: DspCh=0 ;HA ;IP Media:
ExtVoicePrompt=0MB ;Security: MediaEncryption StrongEncryption
EncryptControlProtocol ;DSP Voice features: ;Control
Protocols: MSFT FEU=3 SIPRec=3 WebRTC MGCP SIP SBC=3 ;Default
features;;Coders: G711 G726;. . . . ;MAC Addresses in use:. ;-
----- . ;GROUP_1 - fa:16:3e:5f:9a:64. ;---
----- . . . ."
}
```


7.4 Software Load

The `/files/software` URL provides the ability to modify the device software load. Uploading of the software load triggers a device reset to activate it.

URL

```
/api/v1/files/software
```

HTTP Method

```
PUT
```

HTTP Responses

- 200 OK
- 400 Bad request – provided software load is incorrect.
- 409 Conflict – software load can't be applied due to the current device state (e.g. synchronization with the redundant device is in progress).

Example

```
PUT /api/v1/files/software HTTP/1.1
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=----
WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file";
filename="software.cmp"
Content-Type: application/octet-stream

<cmp file>
-----WebKitFormBoundary7MA4YWxkTrZu0gW--

HTTP/1.1 200 OK
Content-Type: application/json
{
  "description": "Device will reset now to activate new
software load"
}

or

HTTP/1.1 409 Conflict
Content-Type: application/json
{
  "description": "Device is currently performing HA
synchronization"
}
```

7.4.1 Hitless Software Upgrade

The `/files/software/hitless` URL provides the ability to upgrade the software load on an HA system via the “hitless” procedure (without service interruption).

URL

```
/api/v1/files/software/hitless
```

HTTP Method

```
PUT
```

HTTP Responses

- 200 OK
- 400 Bad request – provided software load is incorrect.
- 409 Conflict – software load can’t be applied due to the current device state (e.g. synchronization with the redundant device is in progress).

Example

```
PUT /api/v1/files/software/hitless HTTP/1.1
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=-----WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file";
filename="software.cmp"
Content-Type: application/octet-stream

<cmp file>
-----WebKitFormBoundary7MA4YWxkTrZu0gW--

HTTP/1.1 200 OK
Content-Type: application/json
{
  "description": "Device will perform switchover to activate
new software load"
}

or

HTTP/1.1 409 Conflict
Content-Type: application/json
{
  "description": "Device is currently performing HA
synchronization"
}
```

7.5 Additional Files

Additional files (e.g. auxiliary files) can be loaded to the device using the same mechanism as described in Section 7.4.

The following additional files are supported:

- /files/amd – answering machine detection
- /files/castable – CAS table
- /files/codertable – external coders table
- /files/cpt – call progress tones
- /files/dialplan – dialing plan
- /files/prt – pre-recorded tones
- /files/voiceprompts – voice prompts
- /files/configurationPackage.tar.gz – configuration package
- /files/sbcWizard – SBC wizard template package

URL

```
/api/v1/files/<filename>
```

HTTP Method

```
PUT
```

HTTP Responses

- 200 OK
- 400 Bad request – provided software load is incorrect.
- 409 Conflict – software load can't be applied due to the current device state (e.g. synchronization with the redundant device is in progress).

Example

```
PUT /api/v1/files/cpt HTTP/1.1
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=-----WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file";
filename="tones.cpt"
Content-Type: application/octet-stream

<call progress tones file>
-----WebKitFormBoundary7MA4YWxkTrZu0gW--
```

This page is intentionally left blank.

8 TLS Context Files

The `/files/tls` URL provides access to the device certificates, private key and trusted root certificates of the TLS context.

URL

```
/api/v1/files/tls
```

HTTP Method

```
GET
```

HTTP Response

```
■ 200 OK
```

Example

```
GET /api/v1/files/tls HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "tls": [
    {
      "id": "0",
      "name": "default",
      "url": "/api/v1/files/tls/0"
    }
  ]
}
```



Note: The creation / configuration / removal of TLS contexts should be performed via other APIs – e.g. by uploading incremental ini file or CLI script as described in Sections 7.2 and 7.3. The APIs described in this chapter are for manipulation of “files” associated with existing TLS contexts.

8.1 Selecting TLS Context

The `/files/tls/<id>` URL provides access to the specific TLS context by its `<id>`.

URL

```
/api/v1/files/tls/<id>
```

HTTP Method

```
GET
```

HTTP Responses

■ 200 OK

Example

```
GET /api/v1/files/tls/2 HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "items": [
    {
      "id": "privateKey",
      "description": "Private key",
      "url": "/api/v1/files/tls/2/privateKey"
    },
    {
      "id": "certificate",
      "description": "TLS certificate",
      "url": "/api/v1/files/tls/2/certificate"
    },
    {
      "id": "trustedRoot",
      "description": "Trusted root",
      "url": "/api/v1/files/tls/2/trustedRoot"
    }
  ]
}
```

8.2 Private Key

The `/files/tls/<id>/privateKey` URL provides access to the private key of the specific TLS context. You may verify the size and validity of the current private key or upload a new private key to the device. When uploading (via PUT method), the private key must be specified in PEM format.



Note: In accordance with the best security practices, it is impossible to extract (download) the private key from the device.

URL

```
/api/v1/files/tls/<id>/privateKey
```

HTTP Method

```
GET, PUT
```

Supported Parameters (for PUT request)

Parameter	Type	Description
password	String	(optional) Password of the private key. Default = <none>.

HTTP Responses

- 200 OK
- 400 `Bad request` – provided private key file is incorrect (e.g. not in PEM format or has invalid size).
- 409 `Conflict` – private key can't be loaded due to current device state (e.g. redundant board is synchronizing).

Example 1

```
GET /api/v1/files/tls/2/privateKey HTTP/1.1
```

```
Host: 10.4.219.229
```

```
HTTP/1.1 200 OK
```

```
Content-Type: application/json
```

```
{
  "size": 1024,
  "valid": True // as per "Private Key" status in Web
}
```

Example 2

```
PUT /api/v1/files/tls/2/privateKey HTTP/1.1
```

```
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=----
WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file"; filename="key.pem"
Content-Type: application/octet-stream

-----BEGIN RSA PRIVATE KEY-----
zg1X8vSyH/ED929hjGNF1hAxmIVIgdQdGG3kkWnlmI+4X4kLA3TMHPIkYjwaGPhH
2cdpdkm8KXg8H/hzVIaf/qB6QyiL84d/zRtAG8F1fHVABxkOlSp/kLzHSVT4iD/J
...
YxlA9aGrlI+wsk/h80YF0ly6LwYSfgUaFPdJl1sOjz5bpVTpwT5P0DwT4cPfHRnQ
33Hn3pxbYq22t5Q6r2RE8DEMUAN8gVQ6Ec2JYp901NrQhM4GCHm+mw==
-----END RSA PRIVATE KEY-----
-----WebKitFormBoundary7MA4YWxkTrZu0gW--

HTTP/1.1 200 OK
Content-Type: application/json
{
  "description": "Private key was successfully changed"
}
```

8.2.1 Generate New Private Key

The `/files/tls/<id>/privateKey/generate` URL provides the ability to generate a new private key. The generation occurs on the device and therefore this method is considered to be more secure than the uploading of the private key as described in Section 8.2.

URL

`/api/v1/files/tls/<id>/privateKey/generate`

HTTP Method

POST

Supported Request JSON Attributes

Parameter	Type	Value	Description
size	Number	512, 768, 1024, 2048, 4096	Size of the generated private key. Default = 1024.

HTTP Responses

- 200 OK
- 400 Bad request – provided parameters or values are incorrect
- 409 Conflict – private key can't be generated due to current device state (e.g. redundant board is synchronizing)

Example

```

POST /api/v1/files/tls/2/privateKey/generate HTTP/1.1
Host: 10.4.219.229
Content-Type: application/json
{
  "size": 2048
}

HTTP/1.1 200 OK
Content-Type: application/json
{
  "description": "Private key was successfully generated"
}

```

8.3 Device Certificate

The `/files/tls/<id>/certificate` URL provides access to the device certificate of the specific TLS context. You may download the current certificate or upload a new one. When uploading (via PUT method), the certificate must be specified in PEM format.

URL

```
/api/v1/files/tls/<id>/certificate
```

HTTP Methods

```
GET, PUT
```

HTTP Responses

- 200 OK
- 400 Bad request – provided certificate file is wrong (e.g. not in PEM format)
- 409 Conflict – certificate can't be loaded due to current device state (e.g. redundant board is synchronizing).

Example 1

```

GET /api/v1/files/tls/2/certificate HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/octet-stream
-----BEGIN CERTIFICATE-----

```

```
MIIDMjCCAhqgAwIBAgIBBDANBgkqhkiG9w0BAQUFADAfMQwwCgYDVQQKEwNBQ0wx
DzANBgNVBAMTB1Jvb3RDQTAeFw0wMDAxMDEwMDAwMDBaFw0zMDAxMDEwMDAwMDBa
...
EcqvMKSuAmR8Csl5STrVo+7m4IgeYCTrRZ1hVL/wB8PSD51sg4lGyhos97Q7kH0w
T9cKHStw
-----END CERTIFICATE-----
```

Example 2

```
PUT /api/v1/files/tls/2/certificate HTTP/1.1
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=-----
WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file";
filename="cert.pem"
Content-Type: application/octet-stream

-----BEGIN CERTIFICATE-----
MIIDMjCCAhqgAwIBAgIBBDANBgkqhkiG9w0BAQUFADAfMQwwCgYDVQQKEwNBQ0wx
DzANBgNVBAMTB1Jvb3RDQTAeFw0wMDAxMDEwMDAwMDBaFw0zMDAxMDEwMDAwMDBa
...
EcqvMKSuAmR8Csl5STrVo+7m4IgeYCTrRZ1hVL/wB8PSD51sg4lGyhos97Q7kH0w
T9cKHStw
-----WebKitFormBoundary7MA4YWxkTrZu0gW--

HTTP/1.1 200 OK
Content-Type: application/json
{
  "description": "Certificate was successfully changed"
}
```

8.4 Generate Self-Signed Certificate

The `/files/tls/<id>/certificate/generate` URL generates a new self-signed device certificate for the specific TLS context.

URL

```
/api/v1/files/tls/<id>/certificate/generate
```

HTTP Method

```
POST
```

Supported Request JSON attributes:

Parameter	Type	Description
subjectName	String	Subject name [CN] of the generated certificate. Default = <empty>.
organizationalUnit	String	Organizational unit [OU] of the generated certificate. Default = <empty>.
companyName	String	Company name [O] of the generated certificate. Default = <empty>.
localityName	String	Locality of city name [L] of the generated certificate. Default = <empty>.
state	String	State [ST] of the generated certificate. Default = <empty>.
countryCode	String	Country code [C] of the generated certificate. Default = <empty>.

Supported Responses

- 200 OK
- 400 Bad request - provided certificate file is wrong (e.g. not in PEM format)
- 409 Conflict - private key can't be loaded due to current device state (e.g. redundant board is synchronizing).

Example

```
POST /api/v1/files/tls/2/certificate/generate HTTP/1.1
Host: 10.4.219.229
Content-Type: application/json
{
  "subjectName": "lync-gw.company.com"
}

HTTP/1.1 200 OK
```

```
Content-Type: application/json
{
  "description": "Self-signed certificate was successfully generated"
}
```

8.5 Generate Certificate Signing Request

The `/files/tls/<id>/certificate/generate` URL generates a new certificate signing request (CSR) for the specific TLS context. The generated CSR is returned in the response.

URL

```
/api/v1/files/tls/<id>/certificate/request
```

HTTP Method

```
POST
```

Supported Request JSON attributes

Parameter	Type	Description
subjectName	String	Subject name [CN] of the generated certificate. Default = <empty>.
organizationalUnit	String	Organizational unit [OU] of the generated certificate. Default = <empty>.
companyName	String	Company name [O] of the generated certificate. Default = <empty>.
localityName	String	Locality of city name [L] of the generated certificate. Default = <empty>.
state	String	State [ST] of the generated certificate. Default = <empty>.
countryCode	String	Country code [C] of the generated certificate. Default = <empty>.
signatureAlgorithm	String sha1 sha256 sha512	Signature algorithm to be used for the certificate signing request; default=sha1

Supported Responses

- 200 OK
- 400 Bad request - provided certificate file is incorrect (e.g. it is not in PEM format)

- 409 Conflict – private key can't be loaded due to current device state (e.g. redundant board is synchronizing).

Example

```

POST /api/v1/files/tls/2/certificate/request HTTP/1.1
Host: 10.4.219.229
Content-Type: application/json
{
  "subjectName": "lync-gw.company.com"
}

HTTP/1.1 200 OK
Content-Type: application/octet-stream
-----BEGIN CERTIFICATE REQUEST-----
MIIBZDCBzqIBADAlMRUwEwYDVQQDDAxGQTE2M0VGM0IxREUxDDAKBgNVBAoMA0FD
TDCBnzANBQgqhkiG9w0BAQEFAAOBjQAwwYkCgYEA5sVNvmrwFaPkJUE2zA8TSR78
...
+pa+sF+F+N9HPQ0hqsvBtNJTL5dOEICBYcqYTx5+zqi38WAwHml4VGqduBofZWB2
pEqNck3yG/k8Hmm2pbTUFEE5XpVc6Lcu
-----END CERTIFICATE REQUEST-----

```

8.6 Trusted Root Certificates

The `/files/tls/<id>/trustedRoot` URL provides access to the trusted root store of the specific TLS context. You may download the current content of the store (multiple trusted root certificates) or upload the new content of the store. When uploading (via PUT method), certificates must be specified in PEM format. Multiple certificates may be specified one after another.



Note: This API uploads and downloads complete trusted root store (that may contain multiple certificates). If you need to modify trusted root store by uploading an additional trusted root certificate – use `trustedRoot/incremental` API instead as described in Section 8.7.

URL

```
/api/v1/files/tls/<id>/trustedRoot
```

HTTP Method

```
GET, PUT
```

Supported Responses

- 200 OK
- 400 Bad request – provided certificate file is wrong (e.g. not in PEM format)
- 409 Conflict – private key can't be loaded due to current device state (e.g. redundant board is synchronizing).

Example 1

```
GET /api/v1/files/tls/2/trustedRoot HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/octet-stream

-----BEGIN CERTIFICATE-----
MIIC7jCCAdagAwIBAgIBBjANBgkqhkiG9w0BAQUFADAgMQwwCgYDVQQKEwNBQ0wx
EDA0BgNVBAMUB0NBXzI0MzkwHhcNMDAwMTAxMDAwMDAwWhcNMzAwMTAxMDAwMDAw
...
kedoijcGdGJ9xA0bZa/lFqQQWPnKn735B5d5yjGPStHrh4QgtMaK6x3RmMnuPjoo
nK4zC2nJLBYcTpJU1AQvEFsoiLaBmyJl0wNF8HY3IgcT8g==
-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----
MIIC7jCCAdagAwIBAgIBBTANBgkqhkiG9w0BAQUFADAgMQwwCgYDVQQKEwNBQ0wx
EDA0BgNVBAMUB0NBXzI0MzkwHhcNMDAwMTAxMDAwMDAwWhcNMzAwMTAxMDAwMDAw
...
3PTmpOih9jPFd69pjzgzDef8E3JsmYfQUHiokwnkcpC6od8WRu4JMnE9jQ4cARi
apkJGofjnELCq4ym/JjskqMSBhNpBUz93/xxZ1f25K1XIQ==
-----END CERTIFICATE-----
```

Example 2

```
PUT /api/v1/files/tls/2/trustedRoot HTTP/1.1
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=----
WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file";
filename="trust.pem"
Content-Type: application/octet-stream

-----BEGIN CERTIFICATE-----
MIIC7jCCAdagAwIBAgIBBjANBgkqhkiG9w0BAQUFADAgMQwwCgYDVQQKEwNBQ0wx
EDA0BgNVBAMUB0NBXzI0MzkwHhcNMDAwMTAxMDAwMDAwWhcNMzAwMTAxMDAwMDAw
...
kedoijcGdGJ9xA0bZa/lFqQQWPnKn735B5d5yjGPStHrh4QgtMaK6x3RmMnuPjoo
nK4zC2nJLBYcTpJU1AQvEFsoiLaBmyJl0wNF8HY3IgcT8g==
-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----
MIIC7jCCAdagAwIBAgIBBTANBgkqhkiG9w0BAQUFADAgMQwwCgYDVQQKEwNBQ0wx
EDA0BgNVBAMUB0NBXzI0MzkwHhcNMDAwMTAxMDAwMDAwWhcNMzAwMTAxMDAwMDAw
...
3PTmpOih9jPFd69pjzgzDef8E3JsmYfQUHiokwnkcpC6od8WRu4JMnE9jQ4cARi
apkJGofjnELCq4ym/JjskqMSBhNpBUz93/xxZ1f25K1XIQ==
-----END CERTIFICATE-----
-----WebKitFormBoundary7MA4YWxkTrZu0gW--
```

```

HTTP/1.1 200 OK
Content-Type: application/json
{
  "description": "Trusted root store was successfully
changed"
}

```

8.7 Add Certificate to Trusted Root Store

The `/files/tls/<id>/trustedRoot/incremental` URL adds additional certificate to the trusted root store.

URL

```
/api/v1/files/tls/<id>/trustedRoot/incremental
```

HTTP Method

```
PUT
```

Supported Responses

- 200 OK
- 400 Bad request - provided certificate file is wrong (e.g. not in PEM format)
- 409 Conflict - private key can't be loaded due to current device state (e.g. redundant board is synchronizing).

Example

```

PUT /api/v1/files/tls/2/trustedRoot/incremental HTTP/1.1
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=----
WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file";
filename="trust.pem"
Content-Type: application/octet-stream

-----BEGIN CERTIFICATE-----
MIIC7jCCAdagAwIBAgIBBjANBgkqhkiG9w0BAQUFADAgMQwwCgYDVQQKEwNBQ0wx
EDA0BgNVBAMUB0NBXzI0MzkwHhcNMDAwMTAxMDAwMDAwWhcNMzAwMTAxMDAwMDAw
...
kedoijcGdGJ9xA0bZa/lFqQQWPnKn735B5d5yjGPStHrh4QgtMaK6x3RmMnuPjoo
nK4zC2nJLBYcTpJULAQvEFsoiLaBmyJl0wNF8HY3IgcT8g==
-----END CERTIFICATE-----
-----WebKitFormBoundary7MA4YWxkTrZu0gW--

HTTP/1.1 200 OK

```

```
Content-Type: application/json
{
  "description": "Trusted root certificate was successfully
added"
}
```


9 Alarms

The `/alarms` URL provides the ability to retrieve the device active and history alarms.

URL

```
/api/v1/alarms
```

HTTP Method

```
GET
```

HTTP Response

```
200 OK
```

Example

```
GET /api/v1/alarms HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "alarms": [
    {
      "id": "active",
      "description": "Active alarms",
      "url": "/api/v1/alarms/active"
    },
    {
      "id": "history",
      "description": "History alarms",
      "url": "/api/v1/alarms/history"
    }
  ]
}
```

9.1 Active Alarms

The `/alarms/active` URL provides the ability to retrieve active device alarms.

URL

`/api/v1/alarms/active`

HTTP Method

GET

Supported Parameters

Parameter	Type	Description
?limit=<value>	Number	Limits response to a specified number of alarms. Note that the device may return fewer alarms – e.g. if no more alarms exist or if the user-specified number is too large. Default = 20.
?after=<value>	as returned in previous response	Returns alarms after the alarm specified by the cursor. The cursor value should be taken from “cursor” element in the previous response.
?before=<value>	as returned in previous response	Returns alarms before the alarm specified by the cursor (backwards search). The cursor value should be taken from the “cursor” element in the previous response.

HTTP Responses

- 200 OK
- 204 No Content – when no alarms are found

Example 1

```
GET /api/v1/alarms/active HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "alarms": [
    {
      "id": "1",
      "description": "Trunk is down",
      "url": "/api/v1/alarms/active/1"
    },
    {
      "id": "2",
      "description": "Device will explode in 15 min",
```

```
        "url": "/api/v1/alarms/active/2"
      }
    ],
    "cursor": {
      "after": "2",
      "before": "-1"
    }
  }
}
```

The 200 OK response includes the “cursor” structure that includes “before” and “after” cursors that may be used in consequent requests. Value “-1” indicates than no more alarms before or after exist.

Example 2

```
GET /api/v1/alarms/active?after=2 HTTP/1.1
```

```
Host: 10.4.219.229
```

```
HTTP/1.1 200 OK
```

```
Content-Type: application/json
```

```
{
  "alarms": [
    {
      "id": "3",
      "description": "Intrusion detected",
      "url": "/api/v1/alarms/active/3"
    }
  ],
  "cursor": {
    "after": "-1",
    "before": "3"
  }
}
```

Example 3

```
GET /api/v1/alarms/active?after=3 HTTP/1.1
```

```
Host: 10.4.219.229
```

```
HTTP/1.1 204 No Content
```

9.1.1 Specific Active Alarm

Use the following URL to retrieve a specific active alarm.

URL

`/api/v1/alarms/active/<id>`

HTTP Method

GET

Supported Parameters

Parameter	Type	Description
?oid=<value>	Number	If value 1 is specified, response will include "oid" attribute that indicated OID of the corresponding SNMP trap. Default = 0.

HTTP Responses

- 200 OK
- 404 Not Found – when alarm is not found

Example

```
GET /api/v1/alarms/active/1 HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "id": "1",
  "description": "Trunk is down",
  "severity": "Major",
  "source": "Board#1",
  "date": "2010-03-01T23:00:00.000Z",
  "url": "/api/v1/alarms/active/1"
}
```

9.1.2 Time of the Last Active Alarm

Use the following URL to retrieve the last time when there was a change to the Active alarms table

URL

```
/api/v1/alarms/active/lastChange
```

HTTP Method

```
GET
```

HTTP Responses

- 200 OK
- 404 Not Found – when alarm is not found

The returned value represents the local device time when last active alarm was raised or cleared in RFC3339 format.

Example

```
GET /api/v1/alarms/active/lastChange HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "lastChange": "2016-06-09 19:00:00+03:00"
}
```

9.2 Alarm History

The `/alarms/history` URL provides the ability to retrieve device alarms history, including all alarms raised and cleared by the device since the last reboot.

URL

```
/api/v1/alarms/history
```

HTTP Method

```
GET
```

Supported Parameters

Parameter	Type	Description
?limit=<value>	Number	Limits response to a specified number of alarms. Note that the device may return fewer alarms – e.g. if no more alarms exist or if the user-specified number is too large. Default = 20.
?after=<value>	As returned in previous response	Returns alarms after the alarm specified by the cursor. The cursor value should be taken from the “cursor” element in the previous response.
?before=<value>	As returned in previous response.	Returns alarms before the alarm specified by the cursor (backwards search). The cursor value should be taken from the “cursor” element in the previous response.

HTTP Responses

- 200 OK
- 204 No Content – when no alarms are found

Example

```
GET /api/v1/alarms/history HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "alarms": [
    {
      "id": "1",
      "description": "Trunk is down",
      "url": "/api/v1/alarms/active/1"
    },
    {
      "id": "2",
```

```
        "description": "Device will explode in 15 min",
        "url": "/api/v1/alarms/active/2"
    }
],
"cursor": {
    "after": "2",
    "before": "-1"
}
}
```

The 200 OK response includes a “cursor” structure that includes “before” and “after” cursors that may be used in consequent requests. The value “-1” indicates that no more alarms before or after exist.

9.2.1 Specific History Alarm

Use the following URL to retrieve a specific history alarm.

URL

```
/api/v1/alarms/history/<id>
```

HTTP Method

```
GET
```

Supported Parameters

Parameter	Type	Description
?oid=<value>	Number	If value 1 is specified, response will include "oid" attribute that indicated OID of the corresponding SNMP trap. Default = 0.

HTTP Responses

- 200 OK
- 404 Not Found

Example

```
GET /api/v1/alarms/history/1 HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "id": "1",
  "description": "Trunk is down",
  "severity": "Major",
  "source": "Board#1",
  "date": "2010-03-01T23:00:00.000Z",
  "url": "/api/v1/alarms/history/1"
}
```


10 Device Status

The `/status` URL displays the device status summary.

URL

```
/api/v1/status
```

HTTP Method

```
GET
```

HTTP Response

```
200 OK
```

Example

```
GET /api/v1/status HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "localTimeStamp": "2010-01-17T17:29:15.000Z",
  "ipAddress": "10.4.219.229",
  "subnetMask": "255.255.0.0",
  "defaultGateway": "10.4.0.1",
  "productType": "Mediant SW",
  "versionID": "7.20A.200.014",
  "protocolType": "SIP",
  "operationalState": "UNLOCKED",
  "highAvailability": "Not Operational",
  "serialNumber": "101780235059663",
  "macAddress": "fa163e6e7e1d",
  "systemUpTime": 161446
}
```

This page is intentionally left blank.

11 Performance Monitoring

This section describes the `/kpi` URL, which provides access to performance monitoring parameters (PMs), also known as key performance indicators (KPIs), which are collected by the device.

URL

```
/api/v1/kpi
```

HTTP Method

```
GET
```

HTTP Response

```
200 OK
```

Example

```
GET /api/v1/kpi HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "items": [
    {
      "id": "current",
      "description": "Real-time KPIs",
      "url": "/api/v1/kpi/current"
    },
    {
      "id": "history",
      "description": "Historical KPIs",
      "url": "/api/v1/kpi/history"
    },
    {
      "id": "interval",
      "description": "Intervals Information",
      "url": "/api/v1/kpi/interval"
    }
  ]
}
```

11.1 Hierarchical Tree Structure of KPI

The Performance Monitoring parameters are organized into the following hierarchical tree structure:

```

/api/v1/kpi
  /current
    /<application>
      /<group>
        /global
          /<pm>
            /<element>
              /<id>
                /<pm>
  /interval
    /<num>
      /<application>
        /<group>
          /global
            /<pm>
              /<element>
                /<id>
                  /<pm>

```

Where:

- <application>: Application name (e.g., "sbc", "gateway", or "media")
- <group>: Group name within the specific application (e.g., "callStat")
- <element>: Name of configuration element to which the performance monitoring parameter belongs (e.g., "ipGroup")
- <id>: Index of the configuration element (e.g., IP Group Index 2)
- <pm>: Name of the specific performance monitoring parameter
- <num>: The index number of the collection interval

For the names of the Performance Monitoring parameters, refer to the *SBC-Gateway Performance Monitoring Reference Guide*.

11.2 Cursor-based Pagination

Some responses provide cursor information, allowing you to paginate through the entities/elements/pms/results. It shows what values you can use in your next request to get the next batch of results.

Cursor information is displayed under the "cursor" field and includes the following fields:

- "start": Indicates the first result in the queried URL. This can be an index number to represent, for example, an IP Group, or a string to represent, for example, a performance monitoring parameter.
- "before": Indicates what values you can use in you next request to get the previous batch of results located before the queried URL. This can be an index number to represent, for example, IP Groups (elements), or a string to represent, for example, a performance monitoring parameter. The value "-1" indicates that there are no more results to return.
- "after": Indicates what values you can use in you next request to get the next batch of results located after the queried URL. This can be an index number to represent, for

example, IP Groups (elements), or a string to represent, for example, a performance monitoring parameter. The value "-1" indicates that there are no more results to return.

- "end": Indicates the last result in the queried URL. This can be an index number to represent, for example, an IP Group ID, or a string to represent, for example, a performance monitoring parameter.

Example 1

The below requests the first two IP Groups. The response also includes cursor information, which indicates the following:

- First IP Group is Index 0 ("start": "0")
- No additional IP Groups exists before the displayed ("before": "-1")
- Next value that can be used to get the next batch of results is "1" ("after": "1")
- Last IP Group result is Index 2 ("end": "2")

```
GET /api/v1/kpi/current/sbc/callStats/ipGroup?limit=2 HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "items": [
    {
      "id": "0",
      "name": "Teams PS",
      "description": "Teams server",
      "url":
"/api/v1/kpi/current/sbc/callStats/ipGroup/0"
    },
  ],
  "cursor": [
    {
      "start": "0",
      "before": "-1",
      "after": "1",
      "end": "2"
    }
  ]
}
```

The below example navigates the above query to the next "page" of results:

```
GET /api/v1/kpi/current/sbc/callStats/ipGroup?limit=2&after=1
HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "items": [
    {
```

```

        "id": "2",
        "name": "ITSP",
        "description": "ITSP server",
        "url":
"/api/v1/kpi/current/sbc/callStats/ipGroup/2"
    },
    {
        "id": "1",
        "name": "Teams-c",
        "description": "Teams client",
        "url":
"/api/v1/kpi/current/sbc/callStats/ipGroup/1"
    }
],
"cursor": [
    {
        "start": "0",
        "before": "2",
        "after": "-1",
        "end": "2"
    }
]
}

```

Example 2

The below requests the first two performance monitoring parameters for IP Group ID #0. The response also includes cursor information, which indicates the following:

- First performance monitoring parameter is abnormalTerminatedCallsInTotal
- No additional performance monitoring parameter results exist before the displayed
- Next result is performance monitoring parameter abnormalTerminatedCallsOutTotal
- Last result is performance monitoring parameter shortCallsCounterTotal

GET

```
/api/v1/kpi/current/sbc/callStats/ipGroup/0?detailed=false&limit=2
```

```
HTTP/1.1
```

```
Host: 10.4.219.229
```

```
HTTP/1.1 200 OK
```

```
Content-Type: application/json
```

```

{
  "items": [
    {
      "id": "abnormalTerminatedCallsInTotal",
      "value": "0"
    }
    {
      "id": "abnormalTerminatedCallsOutTotal",
      "value": "0"
    }
  ],

```

```

    "cursor": [
      {
        "start": "abnormalTerminatedCallsInTotal",
        "before": "-1",
        "after": "abnormalTerminatedCallsOutTotal",
        "end": "shortCallsCounterTotal"
      }
    ]
  }
}

```

11.3 Historical Intervals Discovery

Use the following URL to retrieve (discover) information of specific historical collection (measurement) intervals. The response contains a description of all available collection intervals. For each interval, the following attributes are shown:

- id: Interval index
- start: Start time of the collection interval (local device time in RFC 3339 format)
- end: End time of the collection interval (local device time in RFC 3339 format)

URL

```
/api/v1/kpi/interval
```

HTTP Method

```
GET
```

Supported Parameters

Parameter	Type	Description
?id=<Index>	Number	Returns a description of the specified interval number (<Index>).
?id=last	String	Returns a description of the last (most recent) interval.
?before=<Index>	Number	Returns the description of the intervals that occurred before the specified interval.
?after=<Index>	Number	Returns the description of the intervals that occurred after the specified interval.
?limit=<Count>	Number	Returns the description of the last specified number (count) of intervals. For example, to request the last 4 intervals, the Get must be set to "?limit=4".
<Cursor Information>	String	Returns next (<i>after</i>) or previous (<i>before</i>) results (see Section Cursor-based Pagination for more information).

HTTP Responses

- 200 OK
- 204 No Content – no intervals are available

Example 1

```

GET /api/v1/kpi/interval?id=2 HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "intervals": [
    {
      "id": 2,
      "start": "2020-07-14T10:06:00+01:00",
      "end": "2020-07-14T10:07:00+01:00",
      "url": "/api/v1/kpi/interval?id=2"
    }
  ]
}

```

11.4 Application, Group and Entity Discovery

Use the following URL to retrieve the applications, groups or entities (not the actual performance monitoring parameter). The following attributes are specified for the last interval:

- id – node ID
- description – short textual description

On application discovery, groups are also returned per application.

URL

- Application:

```

/api/v1/kpi/current
/api/v1/kpi/history

```

- Group:

```

/api/v1/kpi/current/<app>
/api/v1/kpi/history/<app>

```

- Entity:

```

/api/v1/kpi/current/<app>/<group>
/api/v1/kpi/history/<app>/<group>

```

HTTP Method

```
GET
```

Supported Parameters

Parameter	Type	Description
?before=<Name>	String	Returns the applications, groups or entities that are listed before the specified application, group or entity.

Parameter	Type	Description
?after=<Name>	String	Returns the applications, groups or entities that are listed after the specified application, group or entity. For example, to return a list of all applications listed after the sbc application: <code>/api/v1/kpi/current?after=sbc</code>
?limit=<Count>	Number	Returns the first number (count) of listed applications, groups or entities. For example, to request the first 3, the Get must be set to "?limit=3".
?kpi=<Name>	String	Returns the description and value of a specified performance monitoring parameter (located in the requested path). For example: <code>/api/v1/kpi/current/sbc/callStats/global?kpi=busyCallsInTotal</code>
?detailed=true false	String	When it equals <code>false</code> (default is <code>enabled</code>) some fields are not returned in the response (e.g., name and description).
<Cursor Information>	String	Returns next (<i>after</i>) or previous (<i>before</i>) results (see Section Cursor-based Pagination for more information).

HTTP Responses

- 200 OK
- 204 No Content – nothing to discover
- 400 Bad Request – bad query parameter or invalid path

Example 1

```

GET /api/v1/kpi/current HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "items": [
    {
      "id": "sbc",
      "description": "SBC application statistics",
      "url": "/api/v1/kpi/current/sbc",
      "groups": [
        "callStats",
        "otherStats",
        "sipRecStats"
      ]
    },
    {
      "id": "media",
      "description": "Media application statistics",
      "url": "/api/v1/kpi/current/media",
      "groups": [
        "clusterStats",

```

```

        "coderStats"
        "dspStats",
        "mediaStats"
    },
    ...
]
}

```

Example 2

GET /api/v1/kpi/current/sbc HTTP/1.1

Host: 10.4.219.229

HTTP/1.1 200 OK

Content-Type: application/json

```

{
  "items": [
    {
      "id": "callStats",
      "description": "Call statistics",
      "url": "/api/v1/kpi/current/sbc/callStats"
    },
    {
      "id": "otherStats",
      "description": "Other Dialogs statistics",
      "url": "/api/v1/kpi/current/sbc/otherStats"
    },
    ...
  ]
}

```

Example 3

GET /api/v1/kpi/current/sbc/callstats HTTP/1.1

Host: 10.4.219.229

HTTP/1.1 200 OK

Content-Type: application/json

```

{
  "items": [
    {
      "id": "global",
      "description": "Global call statistics",
      "url": "/api/v1/kpi/current/sbc/callStats/global"
    },
    {
      "id": "ipGroup",
      "description": "Per-IPGroup statistics",
      "url": "/api/v1/kpi/current/sbc/callstats/ipGroup"
    },
  ],
}

```

```

    ...
  ]
}

```

11.5 Entity Index (ID) Discovery

Use the following URL to retrieve the entity index information. The following attributes are specified for the last interval:

- id – node ID
- description – short textual description

URL

```

/api/v1/kpi/current/<app>/<group>/<ent>
/api/v1/kpi/history/<app>/<group>/<ent>

```

HTTP Method

GET

Supported Parameters

Parameter	Type	Description
?before=<Index>	Number	Returns the entities (indexes) before the specified entity index.
?after=<Index>	Number	Returns the entities (indexes) after the specified entity index. For example, to return a list of all IP Groups after Index 0 (i.e., 1, 2, 3, and so on): <code>/api/v1/kpi/current/sbc/callStats/ipGroup?after=0</code>
?limit=<Count>	Number	Returns the first number (count) of listed entities (indexes), starting from Index 0. For example, to request the first 3 IP Groups (i.e., Index 0, 1, and 2): <code>/api/v1/kpi/current/sbc/callStats/ipGroup?limit=3.</code>
id=<Index>	Number	Returns the entity of the specified Index. For example, to return IP Group Index 1: <code>/api/v1/kpi/current/sbc/callStats/ipGroup?id=1</code>
?detailed=true false	String	When it equals <code>false</code> (default is <code>enabled</code>) some fields are not returned in the response (e.g., name and description).
<Cursor Information>	String	Returns next (<code>after</code>) or previous (<code>before</code>) results (see Section Cursor-based Pagination for more information).

HTTP Responses

- 200 OK
- 204 No Content – nothing to discover
- 400 Bad Request – bad query parameter or invalid path

Example

```
GET /api/v1/kpi/current/sbc/callStats/ipGroup?after=0 HTTP/1.1
Host: 10.4.219.229
```

```
HTTP/1.1 200 OK
```

```
Content-Type: application/json
```

```
{
  "items": [
    {
      "id": "1",
      "name": "Teams PS",
      "description": "Teams server",
      "url":
"/api/v1/kpi/current/sbc/callStats/ipGroup/1"
    },
    {
      "id": "2",
      "name": "Teams-c",
      "description": "Teams client",
      "url":
"/api/v1/kpi/current/sbc/callStats/ipGroup/2"
    },
    ...
  ]
}
```

11.6 Discovery of Performance Monitoring Entity

Use the following URLs to retrieve the performance monitoring value of specific entities (indexes). The following attributes are specified for the last interval:

- id – performance monitoring name
- value

URL

- Singular entities:

```
/api/v1/kpi/current/<app>/<group>/<ent>
/api/v1/kpi/history/<app>/<group>/<ent>?interval=<idx>
```

- Indexed entities:

```
/api/v1/kpi/current/<app>/<group>/<ent>/<id>
/api/v1/kpi/history/<app>/<group>/<ent>/<id>?interval=<idx>
```

HTTP Method

```
GET
```

Supported Parameters

Parameter	Type	Description
?before=<KPI Name>	Number	Returns all the performance monitoring parameters (information including values) that are listed before the specified performance monitoring parameter of the entity (index).
?after=<KPI Name>	Number	Returns all the performance monitoring parameters (information including values) that are listed after the specified performance monitoring parameter of the entity (index). For example: /api/v1/kpi/current/sbc/callStats/ipGroup/0 ?after=postDialDelay
?limit=<Count>	Number	Returns the first number (count) of listed performance monitoring parameters of the entity (index).
kpi=<Name>	String	Returns the value and information for the specified performance monitoring parameter of the entity (Index).
?interval=<Interval ID>	Number	Returns all the performance monitoring parameters (information including values) for a specified collection interval of the entity (index).
?detailed=true false	String	When it equals <code>false</code> (default is <code>enabled</code>) some fields are not returned in the response (e.g., name and description).
<Cursor Information>	String	Returns next (<code>after</code>) or previous (<code>before</code>) results (see Section Cursor-based Pagination for more information).

HTTP Responses

- 200 OK
- 204 No Content – nothing to discover
- 400 Bad Request – bad query parameter or invalid path

Example 1

```
GET /api/v1/kpi/current/sbc/callStats/ipGroup/0 HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "items": [
    {
      "id": "abnormalTerminatedCallsInTotal",
      "name": "Abnormal Terminated Calls In Total",
      "description": "Total number of abnormally
terminated inbound calls (after connect)",
      "url":
"/api/v1/kpi/current/sbc/callStats/ipGroup/0/abnormalTerminate
dCallsInTotal",
      "value": "5"
```

```
    },
    {
      "id": "abnormalTerminatedCallsOutTotal",
      "name": "Abnormal Terminated Calls Out Total",
      "description": "Total number of abnormally
terminated outbound calls (after connect)",
      "url":
"/api/v1/kpi/current/sbc/callStats/ipGroup/0/abnormalTerminate
dCallsOutTotal",
      "value": "0"
    },
    ...
  ]
}
```

Example 2

```
GET /api/v1/kpi/current/sbc/callStats/ipGroup/0?limit=1 HTTP/1.1
Host: 10.4.219.229
```

```
HTTP/1.1 200 OK
```

```
Content-Type: application/json
```

```
{
  "items": [
    {
      "id": "abnormalTerminatedCallsInTotal",
      "name": "Abnormal Terminated Calls In Total",
      "description": "Total number of abnormally
terminated inbound calls (after connect)",
      "url":
"/api/v1/kpi/current/sbc/callStats/ipGroup/0/abnormalTerminate
dCallsInTotal",
      "value": "5"
    }
  ]
}
```

Example 3

```
GET /api/v1/kpi/current/sbc/callStats/ipGroup/0?kpi=attemptedCallsRate
Out HTTP/1.1
Host: 10.4.219.229
```

```
HTTP/1.1 200 OK
```

```
Content-Type: application/json
```

```
{
  "items": [
    {
      "id": "attemptedCallsRateOut",
      "name": "Attempted Calls Rate Out",
```

```

        "description": "Rate of attempted outbound calls
        (call attempts per second)",
        "url":
        "/api/v1/kpi/current/sbc/callStats/ipGroup/0/attemptedCallsRateOut",
        "value": "2"
    }
]
}

```

11.7 Specific Performance Monitoring Parameters

Use the following URL to retrieve the specific performance monitoring data.

The following attributes are specified for the last interval:

- id – performance monitoring name
- value

URL

- Singular entities:

```

/api/v1/kpi/current/<app>/<group>/<ent>/<kpi>
/api/v1/kpi/history/<app>/<group>/<ent>/<kpi>?interval=<idx>

```

- Indexed entities:

```

/api/v1/kpi/current/<app>/<group>/<ent>/<id>/<kpi>
/api/v1/kpi/history/<app>/<group>/<ent>/<id>/<kpi>?interval=<idx>

```

HTTP Method

GET

Supported Parameters

Parameter	Type	Description
?interval=<Index>	Number	Returns the performance monitoring parameter's value for the specified interval.
?interval=last	String	Returns the performance monitoring parameter's value for the last (most recent) measurement interval.
?interval=all	String	Returns the performance monitoring parameter's values for all measurement intervals.
?detailed=true false	String	When it equals <code>false</code> (default is <code>enabled</code>) some fields are not returned in the response (e.g., name and description).

HTTP Responses

- 200 OK
- 204 Bad Request – nothing to display
- 404 Not Found – invalid path

Example 1

```
GET /  
/api/v1/kpi/current/sbc/callStats/ipGroup/0/noAnswerCallsInTotal  
HTTP/1.1  
Host: 10.4.219.229  
  
HTTP/1.1 200 OK  
Content-Type: application/json  
{  
  "id": "noAnswerCallsInTotal ",  
  "value": 10  
}
```

Example 2

```
GET /  
/api/v1/kpi/current/sbc/callStats/ipGroup/0/noAnswerCallsInTotal  
?interval=all HTTP/1.1  
Host: 10.4.219.229  
  
HTTP/1.1 200 OK  
Content-Type: application/json  
{  
  "items": [  
    {  
      "interval": "21",  
      "value": "0"  
    },  
    {  
      "interval": "20",  
      "value": "0"  
    },  
    {  
      "interval": "19",  
      "value": "0"  
    }  
  ]  
}
```


12 License Management

The `/license` URL provides the ability to view and modify the device license key.

URL

```
/api/v1/license
```

HTTP Method

```
GET, PUT
```

Request Content Types

PUT command may use one of the following content types:

- `application/json` – see description of Supported Parameters below; not supported for HA configurations
- `form/multi-part` – supported for all configurations; may include multiple license keys and the device will apply the relevant key based on the corresponding serial number. In an HA configuration, the license may be applied to both the active and redundant devices.

Supported Request JSON Attributes

Attribute	Type	Value	Description
licenseVersion	Number	1	License version. Only version 1 is currently supported.
serialNumber	String		Serial number. If specified – compared to the device's serial number and if a mismatch is found, the update request is rejected. This attribute is optional.
key	String		License key in encrypted format.

HTTP Responses

- 200 OK
- 400 Bad request - provided license key is incorrect.
- 409 Conflict – license key can't be loaded due to the current device state (e.g. application/json Content-Type is used for HA device).

Example 1

```
GET /api/v1/license HTTP/1.1
Host: 10.4.219.229

HTTP/1.1 200 OK
Content-Type: application/json
{
  "licenseVersion": 1,
  "serialNumber": "277522263687112",
```

```
  "key": "jCx6r5tovCIKaBBbhPtT53Yj",
  "keyDescription": "Key features: Board Type: Mediant 800
Security: IPSEC MediaEncryption StrongEncryption
EncryptControlProtocol
Coders: G723 G729 G728 NETCODER GSM-FR GSM-EFR AMR EVRC-QCELP
G727 ILBC EVRC-B "
```

Example 2

```
PUT /api/v1/license HTTP/1.1
Host: 10.4.219.229
Content-Type: application/json
{
  "licenseVersion": 1,
  "serialNumber": "277522263687112",
  "key": "jCx6r5tovCIKaBBbhPtT53Yj"
}

HTTP/1.1 200 OK
Content-Type: application/json
{
  "description": "Device will reset now to activate new
license"
}

or

HTTP/1.1 409 Conflict
Content-Type: application/json
{
  "description": "License key can't be applied to device in
HA configuration. Use license file instead."
}
```

Example 3

```
PUT /api/v1/license HTTP/1.1
Host: 10.4.219.229
Content-Type: multipart/form-data; boundary=----
WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file"; filename="key.txt"
Content-Type: application/octet-stream

<license file>
-----WebKitFormBoundary7MA4YWxkTrZu0gW--

HTTP/1.1 200 OK
```

```
Content-Type: application/json
{
  "description": "Device will reset now to activate new
software load"
}
```

This page is intentionally left blank.

13 Full List of Supported HTTP Responses

The following HTTP responses are used by the REST API:

- 200 `OK` – indicates successful request completion.
- 201 `Created` – indicates the creation of a new resource.
- 204 `No Content` – indicates that no items are found in response to a discovery request.
- 400 `Bad Request` – indicates a request failure due to an invalid input.
- 401 `Unauthorized` – indicates a request failure due to incorrect authentication credentials.
- 403 `Forbidden` – indicates a request failure due to an authorization failure (i.e. URL exists; however the user is not authorized to access it).
- 404 `Not Found` – indicates an invalid URL.
- 405 `Method Not Allowed` – indicates that the HTTP method is not supported on the specific URL/resource.
- 406 `Not Acceptable` – indicates that the client included “Accept:” header in a request that doesn’t include the format used by the server (for most URLs it’s “application/JSON”).
- 409 `Conflict` – indicates a failure due to “intermittent” reason (e.g. synchronization with the redundant device is in progress).
- 500 `Internal Server Error` – indicates an internal failure.

International Headquarters

1 Hayarden Street,
Airport City
Lod 7019900, Israel
Tel: +972-3-976-4000
Fax: +972-3-976-4040

AudioCodes Inc.

200 Cottontail Lane,
Suite A101E,
Somerset, NJ 08873
Tel: +1-732-469-0880
Fax: +1-732-469-2298

Contact us: <https://www.audiocodes.com/corporate/offices-worldwide>

Website: <https://www.audiocodes.com/>

©2021 AudioCodes Ltd. All rights reserved. AudioCodes, AC, HD VoIP, HD VoIP Sounds Better, IPmedia, Mediant, MediaPack, What's Inside Matters, OSN, SmartTAP, User Management Pack, VMAS, VoIPerfect, VoIPerfectHD, Your Gateway To VoIP, 3GX, VocaNom, AudioCodes One Voice, AudioCodes Meeting Insights, AudioCodes Room Experience and CloudBond are trademarks or registered trademarks of AudioCodes Limited. All other products or trademarks are property of their respective owners. Product specifications are subject to change without notice.

Document #: LTRT-41768

