



REFERENCE GUIDE

Software 3.8.0 | November2021 | 3725-86572-008A

Poly VideoOS REST API

Poly G7500, Studio X70, Studio X50, and Studio X30

GETTING HELP

For more information about installing, configuring, and administering Poly products or services, go to [Poly Online Support Center](#).

Plantronics, Inc. (Poly – formerly Plantronics and Poly)
345 Encinal Street
Santa Cruz, California
95060

© 2021 Plantronics, Inc. All rights reserved. Poly, the propeller design, and the Poly logo are trademarks of Plantronics, Inc. All other trademarks are the property of their respective owners.

Contents

- Before You Begin 2**
 - Prerequisite Information 2
 - Audience and Purpose of This Guide 2
 - Related Poly and Partner Resources 2
- REST API Commands 4**
 - Commands and Structure 4
 - Audio* 4
 - Calendar* 16
 - Cameras* 22
 - Collaboration* 45
 - Conferences* 51
 - Directory* 78
 - Session* 107
 - System* 108
 - SysTracker* 131
 - Video* 132
- Diagnostics 136**
 - Error Codes 136
 - Logging 137

Before You Begin

This guide describes the REST (Representational State Transfer) API feature, which provides a convenient, scalable, portable, and reliable API for interacting with Poly video systems, enabling users to execute certain functions and retrieve information.

The information in this guide applies to the following Poly video systems:

- Poly G7500
- Poly Studio X70
- Poly Studio X50
- Poly Studio X30

Prerequisite Information

Before you begin working with the REST API, note the following:

- The REST API feature is never disabled.
- Administrator credentials are required for access authentication before REST API commands can be sent. Poly recommends changing the default administrator password before using the REST API.
- All request and response bodies should be processed as JSON formatted data, unless specifically stated otherwise.
- The REST API cannot accept more than 10 MB of data.

Audience and Purpose of This Guide

The primary audience for this manual is systems integrators who intend to enable configuration and management of the system features through integrated systems. This manual is not intended for end users.

Related Poly and Partner Resources

See the following sites for information related to this product.

- The [Poly Online Support Center](#) is the entry point to online product, service, and solution support information including Video Tutorials, Documents & Software, Knowledge Base, Community Discussions, Poly University, and additional services.
- The [Poly Document Library](#) provides support documentation for active products, services, and solutions. The documentation uses responsive HTML5 so you can access and view installation, configuration, or administration content from any online device.
- The [Poly Community](#) provides access to the latest developer and support information. Create an account to access Poly support personnel and participate in developer and support forums. You can find the latest information on hardware, software, and partner solutions topics, share ideas, and solve problems with your colleagues.

- The [Poly Partner Network](#) is an ecosystem of industry leaders who natively integrate the Poly standards-based RealPresence Platform with their customers' current UC infrastructures. Poly Partners make it easy for you to communicate on a human-to-human level with the applications and devices you use every day.
- [Poly Software and Services](#) solutions help your business succeed and get the most out of your investment through the benefits of collaboration.

REST API Commands

This section describes the configuration parameters and API commands for Poly VideoOS, version 3.8.0.

Commands and Structure

Audio

audio

| | |
|-----------------------------------|---|
| Description | This API provides details about the system's audio information. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/audio |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "muteLocked": false, "muted": true, "numOfMicsConnected": 1, "volume": 62 } |
| Applicable return codes | 200 OK, 500 Server Error |

audio

| | |
|-----------------------------------|--|
| Description | This API performs an audio action on the system related to pressed DTMF characters. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/audio |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "action": <string>, "dtmfChar": <string>, "speaker": <string> }</pre> <p><i>action</i> – The action to complete on the system. Valid strings include "start", "playDTMF", "playTestTone", "stopPlayingTestTone", and "itpSegmentDetection".</p> <p><i>dtmfChar</i> – Characters provided as the DTMF keypad input.</p> <p><i>speaker</i> – The speaker selection that plays the tone. Valid strings include "SPEAKER_LEFT", "SPEAKER_RIGHT", and "SPEAKER_BOTH".</p> |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

audio/acousticfence

| | |
|-----------------------------------|---|
| Description | This API performs acoustic fence actions and broadcasts audio fence indicators. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/audio/acousticfence |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "action": <string>, "dtmfChar": <string>, "speaker": <string> }</pre> <p><code>action</code> – The action to complete on the system. Valid strings include “start”, “playDTMF”, “playTestTone”, “stopPlayingTestTone”, and “itpSegmentDetection”.</p> <p><code>dtmfChar</code> – Characters provided as the DTMF keypad input.</p> <p><code>speaker</code> – The speaker selection to play the tone. Valid strings include “SPEAKER_LEFT”, “SPEAKER_RIGHT”, and “SPEAKER_BOTH”.</p> |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

audio/audiometers

| | |
|-----------------------------------|---|
| Description | This API provides system audio meter information. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/audio/audiometers |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | <pre> Body: { "isValid": false, "levelBack": -20, "levelCenter": -20, "levelLeft": -20, "levelRight": -20, "meterIndex": -1, "nameInstance": -1, "portDirection": "IN", "portInstance": 0, "portName": "FAR_IN", "validChanMode": "left_and_right" }, { "isValid": false, "levelBack": -20, "levelCenter": -20, "levelLeft": -20, "levelRight": -20, "meterIndex": -1, "nameInstance": -1, "portDirection": "IN", "portInstance": 1, "portName": "CAMERA_MIC", "validChanMode": "left_and_right" }, { "isValid": false, "levelBack": -20, "levelCenter": -20, "levelLeft": -20, "levelRight": -20, "meterIndex": 1, "nameInstance": 1, "portDirection": "IN", "portInstance": 2, "portName": "IP_TABLE_MIC", "validChanMode": "left_and_right_and_center" }, </pre> |

```
{
  "isValid": false,
  "levelBack": -20,
  "levelCenter": -20,
  "levelLeft": -20,
  "levelRight": -20,
  "meterIndex": 2,
  "nameInstance": 2,
  "portDirection": "IN",
  "portInstance": 3,
  "portName": "POLYCOM_MIC",
  "validChanMode": "left_and_right_and_center"
},
{
  "isValid": false,
  "levelBack": -20,
  "levelCenter": -20,
  "levelLeft": -20,
  "levelRight": -20,
  "meterIndex": 3,
  "nameInstance": 3,
  "portDirection": "IN",
  "portInstance": 4,
  "portName": "POLYCOM_MIC",
  "validChanMode": "left_and_right_and_center"
},
{
  "isValid": false,
  "levelBack": -20,
  "levelCenter": -20,
  "levelLeft": -20,
  "levelRight": -20,
  "meterIndex": 4,
  "nameInstance": 4,
  "portDirection": "IN",
  "portInstance": 5,
  "portName": "POLYCOM_MIC",
  "validChanMode": "left_and_right_and_center"
},
{
  "isValid": false,
  "levelBack": -20,
  "levelCenter": -20,
  "levelLeft": -20,
  "levelRight": -20,
  "meterIndex": -1,
  "nameInstance": -1,
  "portDirection": "IN",
  "portInstance": 6,
  "portName": "MIC_IP_ADAPTER",
  "validChanMode": "left_and_right"
},
{
  "isValid": false,
```

```
"levelBack": -20,
"levelCenter": -20,
"levelLeft": -20,
"levelRight": -20,
"meterIndex": -1,
"nameInstance": -1,
"portDirection": "IN",
"portInstance": 7,
"portName": "CAMERA_MIC",
"validChanMode": "left_and_right"
},
{
  "isValid": true,
  "levelBack": -20,
  "levelCenter": -20,
  "levelLeft": -20,
  "levelRight": -20,
  "meterIndex": -1,
  "nameInstance": -1,
  "portDirection": "IN",
  "portInstance": 9,
  "portName": "THREE_POINT_FIVE",
  "validChanMode": "left_and_right"
},
{
  "isValid": true,
  "levelBack": -20,
  "levelCenter": -20,
  "levelLeft": -20,
  "levelRight": -20,
  "meterIndex": 1,
  "nameInstance": 1,
  "portDirection": "IN",
  "portInstance": 10,
  "portName": "HDMI",
  "validChanMode": "left_and_right"
},
{
  "isValid": true,
  "levelBack": -20,
  "levelCenter": -20,
  "levelLeft": -20,
  "levelRight": -20,
  "meterIndex": -1,
  "nameInstance": -1,
  "portDirection": "OUT",
  "portInstance": 0,
  "portName": "LINE_OUT",
  "validChanMode": "left_and_right"
},
{
  "isValid": false,
  "levelBack": -20,
  "levelCenter": -20,
```

```
    "levelLeft": -20,  
    "levelRight": -20,  
    "meterIndex": -1,  
    "nameInstance": -1,  
    "portDirection": "IN",  
    "portInstance": 11,  
    "portName": "HEADSET_IN",  
    "validChanMode": "left_and_right"  
  },  
  {  
    "isValid": false,  
    "levelBack": -20,  
    "levelCenter": -20,  
    "levelLeft": -20,  
    "levelRight": -20,  
    "meterIndex": -1,  
    "nameInstance": -1,  
    "portDirection": "OUT",  
    "portInstance": 1,  
    "portName": "HEADSET_OUT",  
    "validChanMode": "left_and_right"  
  }  
}
```

| | |
|--------------------------------|--------------------------|
| Applicable return codes | 200 OK, 500 Server Error |
|--------------------------------|--------------------------|

audio/microphones

| | |
|-----------------------------------|---|
| Description | This API provides the list of microphones with relevant information for each one. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/audio/microphones |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <pre>{ "number": 0, "mute": "boolean", "type": 0, "typeInString": "Polycom IP Table Microphone", "hwVersion": "string", "swVersion": "string", "parentID": 0, "state": "MIC_STATE_GREEN" }</pre> |
| Applicable return codes | 200 OK, 500 Server Error |

audio/muted

| | |
|-----------------------------------|---|
| Description | This API provides the microphone mute status. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/audio/muted |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <boolean> |
| Applicable return codes | 200 OK, 500 Server Error |

audio/muted

| | |
|-----------------------------------|--|
| Description | This API performs an audio action on the system to mute or unmute microphones. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/audio/muted |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | Body: <boolean> <boolean> – True mutes the microphones. False unmutes the microphones. |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

audio/volume

| | |
|-----------------------------------|---|
| Description | This API provides information on the microphone volume level. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/audio/volume |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <integer> |
| Applicable return codes | 200 OK, 500 Server Error |

audio/volume

| | |
|-----------------------------------|---|
| Description | This API performs an audio action on the system to set volume level. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/audio/volume |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | Body: <integer> <integer> – An integer representing the volume level. |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

Calendar

calendar

| | |
|-----------------------------------|--|
| Description | This API provides the status of the meetings calendar server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/calendar |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { Status: "ONLINE" } |
| | Valid status strings include DISABLED, OFFLINE, ONLINE, CONNECTING, ERR_AUTHENTICATION, and ERR_UNKNOWN. |
| Applicable return codes | 200 OK, 500 Server Error |

calendar/meetings

| | |
|-----------------------------------|--|
| Description | This API provides a list of meetings in the calendar. The query parameters may specify either a start and end time range, or a maximum number of upcoming events from the current time. The default query returns all upcoming meetings within 24 hours of the current time. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/calendar/meetings |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted:</p> <pre>?number=<integer> ?start=<number>&end=<number></pre> <p>Example Queries:</p> <pre>/rest/calendar/meetings?number=<integer> /rest/calendar/meetings?start=<number>&end=<number></pre> <p><code>number=<integer></code> – An integer representing the maximum number of meeting items to return. Can't be combined with <code>start</code> and <code>end</code> parameters.</p> <p><code>start=<number></code> – The number representing the meeting query start time, in seconds since the epoch (1/1/1970 GMT). The default is the current time.</p> <p><code>end=<number></code> – The number representing the meeting query end time, in seconds since the epoch (1/1/1970 GMT). The default is 24 hours after start time.</p> |
| Output | <p>Body:</p> <pre>{ "actions": ["sip:#####@gatewayaddress.com", "h323:#####@gatewayaddress.com ", ""], "body": "", "canDial": true, "canceled": false, "duration": 3600, "id": "unique meeting ID string", "inCall": false, "location": "location string", "optionalAttendees": [string], "organizer": "organizer string", "privateMeeting": false, "recurring": true, "requiredAttendees": ["unique attendee", "unique attendee"], }</pre> |

```
"startTime": 1572274800,  
"subject": "subject string",  
"lastSyncTime": 1572277393  
}
```

Applicable return codes 200 OK, 500 Server Error

calendar/meetings/{meetingId}

| | |
|-----------------------------------|--|
| Description | This API provides details for a single meeting in the calendar. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/calendar/meetings/<meetingId> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <meetingId> meetingId – The unique meeting identifier to retrieve from the calendar. |
| Output | Body: <pre>{ "actions": ["sip:#####@gatewayaddress.com", "h323:#####@gatewayaddress.com ", ""], "body": "", "canDial": true, "canceled": false, "duration": 3600, "id": "unique meeting ID string", "inCall": false, "location": "location string", "optionalAttendees": [string], "organizer": "organizer string", "privateMeeting": false, "recurring": true, "requiredAttendees": ["unique attendee", "unique attendee"], "startTime": 1572274800, "subject": "subject string", "lastSyncTime": 1572277393 }</pre> |
| Applicable return codes | 200 OK, 500 Server Error |

calendar/meetings/{meetingId}

| | |
|--------------------------------|---|
| Description | This API dials the calendar meeting specified by <code>meetingId</code> . |
| Protocol and Method | Protocol: HTTPS Method: POST Path: <code>/rest/calendar/meetings/<meetingId></code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <code><meetingId></code> <code>meetingId</code> – The unique meeting identifier to retrieve from the calendar. |
| Output | Body: <pre>{ "connectionId": 1, "success": true }</pre> |
| Applicable return codes | 200 OK, 404 Unavailable, 500 Server Error |

calendar/discover

| | |
|-----------------------------------|---|
| Description | This API retrieves information about the calendar server based on the specified e-mail address and credentials. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/calendar/discover |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "domain": <string>, "email": <string>, "username": <string>, "password": <string>, "userdomain": <string> }</pre> <p><code>domain</code> – The domain discovery method. Valid strings include <code>"sip"</code> and <code>"email"</code>. If SIP is configured, the system attempts to discover the calendar server using the system's configured SIP username, versus using the specified e-mail address.</p> <p><code>email</code> – The e-mail address of the account that displays calendaring information.</p> <p><code>username</code> – The user name credential of the account that displays calendaring information.</p> <p><code>password</code> – The password credential of the account that displays calendaring information.</p> <p><code>userdomain</code> – The domain of the account that displays calendaring information.</p> |
| Output | <p>Body:</p> <pre>{ "error": "NOERROR", "serverUrl": "server address string" }</pre> |
| Applicable return codes | 200 OK, 500 Server Error |

Cameras

cameras/sourceselectable

| | |
|-----------------------------------|---|
| Description | This API provides the selection availability of the near source camera. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/cameras/sourceselectable |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: true |
| Applicable return codes | 200 OK, 404 Unavailable, 500 Server Error |

cameras/contentstatus

| | |
|-----------------------------------|---|
| Description | This API provides the system content status. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/cameras/contentstatus |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: "CONTENT_IDLE" |
| Applicable return codes | 200 OK, 500 Server Error |

cameras/near/all

| | |
|-----------------------------------|---|
| Description | This API provides the index number and information for all near camera sources. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/cameras/near/all |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |

Output

Body:

```
{
  "cameraIndex": 1,
  "iconName": null,
  "model": "",
  "name": "Main",
  "sessionID": 0,
  "sourceType": "SRC_PEOPLE",
  "stateAutoFocus": "off",
  "stateFreeze": "off",
  "stateMarker": "off",
  "connected": false,
  "hasAutoFocus": false,
  "hasFocus": false,
  "hasFreeze": false,
  "hasMarker": false,
  "hasPan": false,
  "hasTilt": false,
  "hasZoom": false,
  "nearCamera": true,
  "ptzcapable": false,
  "selected": false,
  "trackable": false
},
{
  "cameraIndex": 2,
  "iconName": null,
  "model": "",
  "name": "Aux",
  "sessionID": 0,
  "sourceType": "SRC_CONTENT",
  "stateAutoFocus": "off",
  "stateFreeze": "off",
  "stateMarker": "off",
  "connected": false,
  "hasAutoFocus": false,
  "hasFocus": false,
  "hasFreeze": false,
  "hasMarker": false,
  "hasPan": false,
```

```
    "hasTilt": false,
    "hasZoom": false,
    "nearCamera": true,
    "ptzcapable": false,
    "selected": false,
    "trackable": false
  },
  {
    "cameraIndex": 3,
    "iconName": null,
    "model": "EAGLEEYE_CUBE_USB",
    "name": "EagleEye Cube USB",
    "sessionID": 0,
    "sourceType": "SRC_PEOPLE",
    "stateAutoFocus": "off",
    "stateFreeze": "off",
    "stateMarker": "off",
    "connected": true,
    "hasAutoFocus": false,
    "hasFocus": false,
    "hasFreeze": false,
    "hasMarker": false,
    "hasPan": true,
    "hasTilt": true,
    "hasZoom": true,
    "nearCamera": true,
    "ptzcapable": true,
    "selected": true,
    "trackable": true
  }
}
```

| | |
|--------------------------------|--------------------------|
| Applicable return codes | 200 OK, 500 Server Error |
|--------------------------------|--------------------------|

cameras/near/people

| | |
|-----------------------------------|---|
| Description | This API provides the index number and information for all near people camera sources. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/cameras/near/people |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <pre>[{ "cameraIndex": 1, "iconName": null, "model": "", "name": "Main", "sessionID": 0, "sourceType": "SRC_PEOPLE", "stateAutoFocus": "off", "stateFreeze": "off", "stateMarker": "off", "connected": false, "hasAutoFocus": false, "hasFocus": false, "hasFreeze": false, "hasMarker": false, "hasPan": false, "hasTilt": false, "hasZoom": false, "nearCamera": true, "ptzcapable": false, "selected": false, "trackable": false }, { "cameraIndex": 3, "iconName": null, "model": "EAGLEEYE_CUBE_USB", "name": "EagleEye Cube USB", "sessionID": 0, "sourceType": "SRC_PEOPLE", "stateAutoFocus": "off", "stateFreeze": "off", "stateMarker": "off", "connected": true, "hasAutoFocus": false, "hasFocus": false, "hasFreeze": false, "hasMarker": false,</pre> |

```
    "hasPan": true,  
    "hasTilt": true,  
    "hasZoom": true,  
    "nearCamera": true,  
    "ptzcapable": true,  
    "selected": true,  
    "trackable": true  
  }  
]
```

| | |
|--------------------------------|--------------------------|
| Applicable return codes | 200 OK, 500 Server Error |
|--------------------------------|--------------------------|

cameras/near/content

| | |
|-----------------------------------|--|
| Description | This API provides the index number and information for all near content camera sources. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/cameras/near/content |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <pre>[{ "cameraIndex": 3, "iconName": null, "model": "EAGLEEYE_CUBE_USB", "name": "EagleEye Cube USB", "sessionID": 0, "sourceType": "SRC_PEOPLE", "stateAutoFocus": "off", "stateFreeze": "off", "stateMarker": "off", "connected": true, "hasAutoFocus": false, "hasFocus": false, "hasFreeze": false, "hasMarker": false, "hasPan": true, "hasTilt": true, "hasZoom": true, "nearCamera": true, "ptzcapable": true, "selected": true, "trackable": true }]</pre> |
| Applicable return codes | 200 OK, 500 Server Error |

cameras/near/{sourceID}

| | |
|-----------------------------------|---|
| Description | This API provides information for the near camera specified by <code>sourceID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: <code>/rest/cameras/near/<sourceID></code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <code><sourceID></code> <code>sourceID</code> – The unique identifier for the camera index. |
| Output | Body: <pre>{ "cameraIndex": 2, "iconName": null, "model": "", "name": "Aux", "sessionID": 0, "sourceType": "SRC_CONTENT", "stateAutoFocus": "off", "stateFreeze": "off", "stateMarker": "off", "connected": false, "hasAutoFocus": false, "hasFocus": false, "hasFreeze": false, "hasMarker": false, "hasPan": false, "hasTilt": false, "hasZoom": false, "nearCamera": true, "ptzcapable": false, "selected": false, "trackable": false }</pre> |
| Applicable return codes | 200 OK, 404 Unavailable, 500 Server Error |

cameras/near/{sourceID}

| | |
|-----------------------------------|---|
| Description | This API performs the move operation for the near camera specified by sourceID. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/cameras/near/<sourceID> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <sourceID></p> <p>sourceID – The unique identifier for the camera index.</p> <p>Body:</p> <pre>{ "action": <string>, "direction": <string>, "message": <string>, "withImage": <string> }</pre> <p>action – Controls either the selected near people camera or controls the near people camera preset. Valid strings are "moveStart", "moveStop", "store", "clear", and "activate".</p> <p>direction – Defines the camera direction for action moveStart. Valid strings are "MOVE_LEFT", "MOVE_RIGHT", "MOVE_UP", "MOVE_DOWN", "MOVE_ZOOMIN", "MOVE_ZOOMOUT", "MOVE_FOCUSNEAR", and "MOVE_FOCUSFAR".</p> <p>message – Defines the camera to select. Valid strings are "LeftCamera", "RightCamera", and "BothCamera".</p> <p>withImage – Defines whether or not to store camera presets with an image. Valid strings are "Yes" and "No".</p> |
| Output | None |
| Applicable return codes | 200 OK, 404 Unavailable, 500 Server Error |

cameras/near/selectedpeople

| | |
|-----------------------------------|--|
| Description | This API provides information for the selected near people camera source. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/cameras/near/selectedpeople |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <pre>{ "cameraIndex": 3, "iconName": null, "model": "EAGLEEYE_CUBE_USB", "name": "EagleEye Cube USB", "sessionID": 0, "sourceType": "SRC_PEOPLE", "stateAutoFocus": "off", "stateFreeze": "off", "stateMarker": "off", "connected": true, "hasAutoFocus": false, "hasFocus": false, "hasFreeze": false, "hasMarker": false, "hasPan": true, "hasTilt": true, "hasZoom": true, "nearCamera": true, "ptzcapable": true, "selected": true, "trackable": true }</pre> |
| Applicable return codes | 200 OK, 500 Server Error |

cameras/near/selectedpeople

| | |
|-----------------------------------|---|
| Description | This API performs the move operation for the selected near people camera source. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/cameras/near/selectedpeople |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "action": <string>, "direction": <string>, "message": <string>, "withImage": <string> }</pre> <p>action – Controls either the selected near people camera or controls the near people camera preset. Valid strings are "moveStart", "moveStop", "store", "clear", and "activate".</p> <p>direction – Defines the camera direction for action moveStart. Valid strings are "MOVE_LEFT", "MOVE_RIGHT", "MOVE_UP", "MOVE_DOWN", "MOVE_ZOOMIN", "MOVE_ZOOMOUT", "MOVE_FOCUSNEAR", and "MOVE_FOCUSFAR".</p> <p>message – Defines the camera to select. Valid strings are "LeftCamera", "RightCamera", and "BothCamera".</p> <p>withImage – Defines whether or not to store camera presets with an image. Valid strings are "Yes" and "No".</p> |
| Output | None |
| Applicable return codes | 200 OK, 500 Server Error |

cameras/near/selectedcontent

| | |
|-----------------------------------|---|
| Description | This API provides information for the selected near content camera source. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/cameras/near/selectedcontent |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <pre> { "cameraIndex": 2, "iconName": null, "model": "", "name": "Aux", "sessionID": 0, "sourceType": "SRC_CONTENT", "stateAutoFocus": "off", "stateFreeze": "off", "stateMarker": "off", "connected": false, "hasAutoFocus": false, "hasFocus": false, "hasFreeze": false, "hasMarker": false, "hasPan": false, "hasTilt": false, "hasZoom": false, "nearCamera": true, "ptzcapable": false, "selected": false, "trackable": false } </pre> |
| Applicable return codes | 200 OK, 500 Server Error |

cameras/near/selectedcontent

| | |
|-----------------------------------|---|
| Description | This API performs the move operation for the selected near content camera source. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/cameras/near/selectedcontent |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "action": <string>, "direction": <string>, "message": <string>, "withImage": <string> }</pre> <p>action – Controls either the selected near people camera or controls the near people camera preset. Valid strings are "moveStart", "moveStop", "store", "clear", and "activate".</p> <p>direction – Defines the camera direction for action moveStart. Valid strings are "MOVE_LEFT", "MOVE_RIGHT", "MOVE_UP", "MOVE_DOWN", "MOVE_ZOOMIN", "MOVE_ZOOMOUT", "MOVE_FOCUSNEAR", and "MOVE_FOCUSFAR".</p> <p>message – Defines the camera to select. Valid strings are "LeftCamera", "RightCamera", and "BothCamera".</p> <p>withImage – Defines whether or not to store camera presets with an image. Valid strings are "Yes" and "No".</p> |
| Output | None |
| Applicable return codes | 200 OK, 500 Server Error |

cameras/near/presets/all

| | |
|-----------------------------------|---|
| Description | This API provides the list of near camera presets. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/cameras/near/presets/all |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <pre>[{ "imageLocation": <string>, "index": 1, "sessionId": 0, "near": true, "stored": true }, { "imageLocation": <string>, "index": 2, "sessionId": 0, "near": true, "stored": true }, { "imageLocation": <string>, "index": 3, "sessionId": 0, "near": true, "stored": true }]</pre> |
| Applicable return codes | 200 OK, 500 Server Error |

cameras/near/presets/{index}

| | |
|-----------------------------------|---|
| Description | This API provides the near camera presets information for a specified camera stored in the index. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/cameras/near/presets/<index> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <index> index – The unique identifier of the near camera to retrieve from the index. |
| Output | Body: <pre>{ "imageLocation": <string>, "index": 1, "sessionId": 0, "near": true, "stored": true }</pre> |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

cameras/near/presets/{index}

| | |
|-----------------------------------|--|
| Description | This API updates the near camera presets information for a specified camera stored in the index. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/cameras/near/presets/<index> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted:</p> <p><index></p> <p>index – The unique identifier of the near camera to retrieve from the index.</p> <p>Body:</p> <pre>{ "action": <string>, "direction": <string>, "message": "string", "withImage": <string> }</pre> <p>action – Controls either the selected near people camera or controls the near people camera preset. Valid strings are "moveStart", "moveStop", "store", "clear", and "activate".</p> <p>direction – Defines the camera direction for action moveStart. Valid strings are "MOVE_LEFT", "MOVE_RIGHT", "MOVE_UP", "MOVE_DOWN", "MOVE_ZOOMIN", "MOVE_ZOOMOUT", "MOVE_FOCUSNEAR", and "MOVE_FOCUSFAR".</p> <p>message – Defines the camera to select. Valid strings are "LeftCamera", "RightCamera", and "BothCamera".</p> <p>withImage – Defines whether or not to store camera presets with an image. Valid strings are "Yes" and "No".</p> |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

cameras/near/position/{sourceID}

| | |
|-----------------------------------|---|
| Description | This API performs the move operation for the selected near camera source, using the exact given coordinates or the coordinates as an offset from the current position. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/cameras/near/position/<sourceID> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <sourceID></p> <p>sourceID – The available value is SELECTED_PEOPLE.</p> <p>Body:</p> <pre>{ "camPosition": { "pan": <integer>, "tilt": < integer>, "zoom": <integer>, }, "relative": <boolean> }</pre> <p>pan – Pan coordinate ranges from -50000 to +50000.</p> <p>tilt – Tilt coordinate ranges from -50000 to +50000.</p> <p>zoom – Zoom coordinate ranges from -50000 to +50000.</p> <p>relative – True sets the given coordinates. False sets an offset of the given coordinates from current position.</p> |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

cameras/far/all

| | |
|-----------------------------------|--|
| Description | This API provides the list and information for all far camera sources. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/cameras/far/all |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | <pre> Body: [{ "cameraIndex": 1, "iconName": null, "model": "", "name": "Main", "sessionID": 0, "sourceType": "SRC_PEOPLE", "stateAutoFocus": "off", "stateFreeze": "off", "stateMarker": "off", "connected": false, "hasAutoFocus": false, "hasFocus": false, "hasFreeze": false, "hasMarker": false, "hasPan": false, "hasTilt": false, "hasZoom": false, "nearCamera": true, "ptzcapable": false, "selected": false, "trackable": false }] </pre> |
| Applicable return codes | 200 OK, 404 Unavailable, 500 Server Error |

cameras/far/selectedpeople

| | |
|-----------------------------------|--|
| Description | This API provides information for the selected far people camera source. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/cameras/far/selectedpeople |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <pre> { "cameraIndex": 3, "iconName": null, "model": "EAGLEEYE_CUBE_USB", "name": "EagleEye Cube USB", "sessionID": 0, "sourceType": "SRC_PEOPLE", "stateAutoFocus": "off", "stateFreeze": "off", "stateMarker": "off", "connected": true, "hasAutoFocus": false, "hasFocus": false, "hasFreeze": false, "hasMarker": false, "hasPan": true, "hasTilt": true, "hasZoom": true, "nearCamera": true, "ptzcapable": true, "selected": true, "trackable": true } </pre> |
| Applicable return codes | 200 OK, 500 Server Error |

cameras/far/selectedpeople

| | |
|-----------------------------------|---|
| Description | This API performs the move operation for the selected far people camera source. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/cameras/far/selectedpeople |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "action": <string>, "direction": <string>, "message": <string>, "withImage": <string> }</pre> <p>action – Controls either the selected near people camera or controls the near people camera preset. Valid strings are "moveStart", "moveStop", "store", "clear", and "activate".</p> <p>direction – Defines the camera direction for action moveStart. Valid strings are "MOVE_LEFT", "MOVE_RIGHT", "MOVE_UP", "MOVE_DOWN", "MOVE_ZOOMIN", "MOVE_ZOOMOUT", "MOVE_FOCUSNEAR", and "MOVE_FOCUSFAR".</p> <p>message – Defines the camera to select. Valid strings are "LeftCamera", "RightCamera", and "BothCamera".</p> <p>withImage – Defines whether or not to store camera presets with an image. Valid strings are "Yes" and "No".</p> |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

cameras/far/{sourceID}

| | |
|-----------------------------------|---|
| Description | This API provides information for the far camera specified by <code>sourceID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: <code>/rest/cameras/far/<sourceID></code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <code><sourceID></code> <code>sourceID</code> – The unique identifier for the camera index. |
| Output | Body: <pre>{ "cameraIndex": 2, "iconName": null, "model": "", "name": "Aux", "sessionID": 0, "sourceType": "SRC_CONTENT", "stateAutoFocus": "off", "stateFreeze": "off", "stateMarker": "off", "connected": false, "hasAutoFocus": false, "hasFocus": false, "hasFreeze": false, "hasMarker": false, "hasPan": false, "hasTilt": false, "hasZoom": false, "nearCamera": true, "ptzcapable": false, "selected": false, "trackable": false }</pre> |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

cameras/far/{sourceID}

| | |
|-----------------------------------|--|
| Description | This API performs the move operation for the far camera specified by <code>sourceID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: <code>/rest/cameras/far/<sourceID></code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <code><sourceID></code></p> <p><code>sourceID</code> – The unique identifier for the camera index.</p> <p>Body:</p> <pre>{ "action": <string>, "direction": <string>, "message": <string>, "withImage": <string> }</pre> <p><code>action</code> – Controls either the selected near people camera or controls the near people camera preset. Valid strings are "moveStart", "moveStop", "store", "clear", and "activate".</p> <p><code>direction</code> – Defines the camera direction for action <code>moveStart</code>. Valid strings are "MOVE_LEFT", "MOVE_RIGHT", "MOVE_UP", "MOVE_DOWN", "MOVE_ZOOMIN", "MOVE_ZOOMOUT", "MOVE_FOCUSNEAR", and "MOVE_FOCUSFAR".</p> <p><code>message</code> – Defines the camera to select. Valid strings are "LeftCamera", "RightCamera", and "BothCamera".</p> <p><code>withImage</code> – Defines whether or not to store camera presets with an image. Valid strings are "Yes" and "No".</p> |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

cameras/snapshot/{snapshotID}

| | |
|-----------------------------------|--|
| Description | This API returns a stored snapshot specified by <code>snapshotID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: <code>/rest/cameras/snapshot/<snapshotID></code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <code><snapshotID></code> <code>snapshotID</code> – The available values are <code>near_image_1.jpg</code> , <code>far_image_1.jpg</code> , or <code>content_image_1.jpg</code> . |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

Collaboration

collaboration

| | |
|-----------------------------------|---|
| Description | This API provides the current state of the collaboration session. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/collaboration |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "sessionId": <string>, "sessionState": "ACTIVE" } |
| Applicable return codes | 200 OK, 404 Unavailable, 500 Server Error |

collaboration

| | |
|-----------------------------------|--|
| Description | This API performs an action on the active collaboration session. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/collaboration |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "action": <string>, }</pre> <p><code>action</code> – The action to perform on the collaboration session. The only valid string is "END".</p> |
| Output | None |
| Applicable return codes | 200 OK, 204 No Content, 400 Bad Request, 500 Server Error |

collaboration/content

| | |
|-----------------------------------|--|
| Description | This API returns a context ID and begins the content save process. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/collaboration/content |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: ?pin=<string></p> <p>Example Query: /rest/collaboration/content?pin=<string></p> <p>pin=<string> – The room pin to authorize access to content session</p> |
| Output | <p>Body:</p> <pre>{ "context_id": "1572362187453" }</pre> |
| Applicable return codes | 200 OK, 204 No Content, 403 Forbidden, 500 Server Error |

collaboration/content

| | |
|-----------------------------------|--|
| Description | This API provides previously-saved content as an attachment, using <code>context_id</code> from the <code>collaboration/content</code> POST command. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: <code>/rest/collaboration/content</code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <code>?pin=<string>&context-id=<string></code></p> <p>Example Query: <code>/rest/collaboration/content?pin=<string>&context-id=<string></code></p> <p><code>pin=<string></code> – The room pin to authorize access to content session.</p> <p><code>context-id=<string></code> – The <code>context_id</code> provided from the <code>collaboration/content</code> POST command.</p> |
| Output | None |
| Applicable return codes | 200 OK, 204 No Content, 500 Server Error |

collaboration/contentssavestatus

| | |
|-----------------------------------|--|
| Description | This API provides the content save enable/disable status. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/collaboration/contentssavestatus |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: ?pin=<string>&context-id=<string></p> <p>Example Query: /rest/collaboration/contentssavestatus?pin=<string>&status=<string></p> <p>pin=<string> – The room pin to authorize access to content session.</p> <p>status=<string> – True enables save. False disables save.</p> |
| Output | None |
| Applicable return codes | 200 OK, 500 Server Error |

collaboration/content/all

| | |
|-----------------------------------|---|
| Description | This API provides and downloads a ZIP file containing saved content from the active collaboration session. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/collaboration/content/all |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: ?pin=<string></p> <p>Example Query: /rest/collaboration/content/all?pin=<string></p> <p>pin=<string> – The room pin to authorize access to content session.</p> |
| Output | None |
| Applicable return codes | 200 OK, 204 No Content, 403 Forbidden, 500 Server Error |

Conferences

conferences

| | |
|-----------------------------------|---|
| Description | This API provides a list of all current conference calls on the system. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/conferences |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | <p>Body:</p> <pre>[{ "capabilities": { "canAcquireChair": false, "canAcquireFloor": false, "canAddTerminal": false, "canAddVideo": false, "canBlastDial": false, "canEscalateCall": false, "canGetRoster": false, "canHangupConference": true, "canHoldConf": false, "canJoinCCCP": false, "canJoinChairControlConf": false, "canJoinInternalConf": false, "canJoinMRC": false, "canLeaveConference": true, "canMuteConference": false, "canMuteTerminal": false, "canPresent": true, "canRemoveTerminal": true, "canShowCloseWide": false, "canSupportSecurityClassification": false, "cannotJoinConf": false }, "connections": [{ "address": "#####@gatewayaddress.com", "answerable": false, "authState": "NONE", "callInfo": "SIP", "callType": "SIP", "canAddVideo": false, "causeCode": 0, "duration": 16, "encrypted": true,</pre> |

```
        "groupname": "",
        "grouptype": "NONE",
        "id": "2",
        "incoming": false,
        "mediaCount": 0,
        "mediaType": "AUDIOVIDEO",
        "parentConfId": "0",
        "percentConnected": 100,
        "rate": 2048,
        "referredBy": "",
        "startTime": 1572444157000,
        "state": "CONNECTED",
        "terminals": [
            {
                "href": "/rest/conferences/0/terminals/",
                "rel": "item"
            }
        ],
        "type": "MCU",
        "videoEscalationState": "NONE"
    }
],
"duration": 16,
"holdStartTime": 0,
"holdState": "NOTHELD",
"id": "0",
"isActive": true,
"isHolding": false,
"isMute": false,
"isSvcConference": false,
"isWaitingInLobby": false,
"mediaServerControlEvent": "MediaServerUnknown",
"mediaServerType": "MediaUnknown",
"startTime": 1572444157000,
"terminals": [
{
    "address": "#####@gatewayaddress.com ",
    "callerID": "#####@gatewayaddress.com ",
    "canSupportMediaStatus": false,
    "fullDescription": false,
    "holdState": "NOTHELD",
    "id": "1",
    "mediaServerType": "MediaUnknown",
    "muteLocked": false,
    "muted": false,
    "name": "participant_name",
    "parentConfId": "0",
    "parentConnectionId": "2",
    "streamsState": "AVC",
    "systemID": "bridge_identification",
    "termType": "REAL"
}
]
}
```

]

Applicable return codes 200 OK, 500 Server Error

conferences

| | |
|-----------------------------------|--|
| Description | This API performs a single participant conference call. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/conferences |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "address": <string>, "rate": <integer>, "dialType": <string>, "password": <string> }</pre> <p>address – The address, hostname, or URI to dial.</p> <p>rate – An integer representing the call bitrate.</p> <p>dialType – The type of call specified in a conference dial request. Not all types may be supported on a given system. Valid strings include "LOOPBACK", "AUTO", "VIDEO", "VOICE", "H320", "H323", "SIP", "GATEWAY", "POTS", "ISDN_VOICE", "SIP_SPEAKERPHONE", "POTS_SPEAKERPHONE", "VOICE_H323", "VOICE_SIP", and "VOICE_GATEWAY".</p> <p>password – The conference password, when required to connect to the conference call.</p> |
| Output | <p>Body:</p> <pre>[{ "rel": "string", "href": "string" }]</pre> |
| Applicable return codes | 200 OK, 415 Error |

conferences/blast

| | |
|-----------------------------------|---|
| Description | This API performs a blast dial to establish a multiple participant conference call. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/conferences/blast |
| Input and Output Syntax | JSON |
| Access Level | Admin |

Input

```

Body:
{
  "endPointList": [
    {
      "addresses": [
        {
          "address": <string>,
          "rate": <integer>,
          "dialType": <string>
        },
        {
          "address": <string>,
          "rate": <integer>,
          "dialType": <string>
        }
      ]
    },
    {
      "addresses": [
        {
          "address": <string>,
          "rate": <integer>,
          "dialType": <string>
        },
        {
          "address": <string>,
          "rate": <integer>,
          "dialType": <string>
        }
      ]
    }
  ],
  "password": <string>
}

```

address – The address, hostname, or URI to dial.

rate – An integer representing the call bitrate.

dialType – The type of call specified in a conference dial request. Not all types may be supported on a given system. Valid strings include "LOOPBACK", "AUTO", "VIDEO", "VOICE", "H320", "H323", "SIP", "GATEWAY", "POTS",


```
"ISDN_VOICE", "SIP_SPEAKERPHONE", "POTS_SPEAKERPHONE",  
"VOICE_H323", "VOICE_SIP", and "VOICE_GATEWAY".
```

`password` – The conference password, when required to connect to the conference call.

| | |
|--------------------------------|---|
| Output | Body: { "rel": "string", "href": "string" } |
| Applicable return codes | 200 OK, 400 Bad Request, 415 Error |

conferences/capabilities

| | |
|-----------------------------------|---|
| Description | This API provides the system's conference call capabilities. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/conferences/capabilities |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "canBlastDial": true, "canMakeAudioCall": true, "canMakeVideoCall": true } |
| Applicable return codes | 200 OK |

conferences/{confID}

| | |
|-----------------------------------|---|
| Description | This API provides information for the specified conference call. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/conferences/<confID> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <confID> confID – The unique identifier of the conference to retrieve. |
| Output | <p>Body:</p> <pre>{ "id": "string", "connections": [{ "id": "string", "type": "ENDPOINT", "state": "INACTIVE", "callType": "UNKNOWN", "mediaType": "UNKNOWN", "callInfo": "string", "causeCode": 0, "rate": 0, "address": "string", "percentConnected": 0, "incoming": true, "encrypted": true, "authState": "NONE", "mediaCount": 0, "terminals": [{ "rel": "self", "href": "string" }] }, "startTime": 0, "duration": 0, "parentConfId": "string", "answerable": true, "groupname": "string", "grouptype": "string", "canAddVideo": true, "videoEscalationState": "string", "referredBy": "string"] }, "terminals": [</pre> |

```
{
  "id": "string",
  "name": "string",
  "callerID": "string",
  "address": "string",
  "systemID": "string",
  "muted": true,
  "holdState": "NOTHELD",
  "termType": "REAL",
  "muteLocked": true,
  "canSupportMediaStatus": true,
  "parentConfId": "string",
  "parentConnectionId": "string",
  "fullDescription": true,
  "mediaServerType": "string",
  "streamsState": "string"
}
],
"capabilities": {
  "canGetRoster": true,
  "canMuteConference": true,
  "canMuteTerminal": true,
  "canPresent": true,
  "canAddTerminal": true,
  "canRemoveTerminal": true,
  "canAcquireChair": true,
  "canAcquireFloor": true,
  "canHangupConference": true,
  "canLeaveConference": true,
  "canBlastDial": true,
  "canEscalateCall": true,
  "canHoldConf": true,
  "canShowCloseWide": true,
  "canJoinChairControlConf": true,
  "canJoinCCCP": true,
  "canJoinMRC": true,
  "canJoinInternalConf": true,
  "cannotJoinConf": true,
  "canSupportSecurityClassification": true,
  "canAddVideo": true
},
"startTime": 0,
"duration": 0,
"isSvcConference": true,
"isActive": true,
"isHolding": true,
"isMute": true,
"holdState": "NOTHELD",
"holdStartTime": 0,
"mediaServerType": "string",
"mediaServerControlEvent": "string",
"isWaitingInLobby": true
}
```

Applicable return codes 200 OK, 400 Bad Request, 404 Unavailable

conferences/{confID}

| | |
|-----------------------------------|---|
| Description | This API hangs up and disconnects the specified conference call. |
| Protocol, Method, and Path | Protocol: HTTPS Method: DELETE Path: /rest/conferences/<confID> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <confID> <confID> – The unique identifier of the conference to retrieve. |
| Output | Body: 0 |
| Applicable return codes | 200 OK, 404 Unavailable, 500 Server Error |

conferences/{confID}/connections

| | |
|-----------------------------------|--|
| Description | This API provides a list of connections associated to the specified conference. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/conferences/<confID>/connections |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: ?filter=<string></p> <p>Example Query: /rest/conferences/<confID>/connections?filter=<string></p> <p>confID – The unique identifier of the conference to retrieve.</p> <p>?filter=<string> – Available values are ALL, ACTIVE, ANSWERABLE, INCOMING, and PENDING.</p> |
| Output | <p>Body:</p> <pre>[{ "address": "#####@gatewayaddress.com", "answerable": false, "authState": "NONE", "callInfo": "SIP", "callType": "SIP", "canAddVideo": false, "causeCode": 0, "duration": 16, "encrypted": true, "groupname": "", "grouptype": "NONE", "id": "2", "incoming": false, "mediaCount": 0, "mediaType": "AUDIOVIDEO", "parentConfId": "0", "percentConnected": 100, "rate": 2048, "referredBy": "", "startTime": 1572444157000, "state": "CONNECTED", "terminals": [{ "href": "/rest/conferences/0/terminals/", "rel": "item" }], "type": "MCU",</pre> |

```
        "videoEscalationState": "NONE"  
    }  
]
```

Applicable return codes 200 OK, 404 Unavailable

conferences/{confID}/connections

| | |
|-----------------------------------|--|
| Description | This API dials and adds a new endpoint connection to an active conference specified by <code>confID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: <code>/rest/conferences/<confID>/connections</code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <code>?filter=<string></code></p> <p>Example Query: <code>/rest/conferences/<confID>/connections</code></p> <p><code>confID</code> – The unique identifier of the conference to retrieve.</p> <p>Body:</p> <pre>{ "address": "string", "rate": 0, "dialType": "LOOPBACK", "password": "string" }</pre> |
| Output | <p>Body:</p> <pre>[{ "rel": "self", "href": "string" }]</pre> |
| Applicable return codes | 200 OK, 404 Unavailable, 415 Error |

conferences/{confID}/connections/{connID}

| | |
|-----------------------------------|--|
| Description | This API provides information for a specific connection using <code>connID</code> , on the specified conference call using <code>confID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: <code>/rest/conferences/<confID>/connections/<connID></code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted:</p> <p><code><confID>/connections/<connID></code></p> <p><code>confID</code> – The unique identifier of the conference to retrieve.</p> <p><code>connID</code> – The unique identifier for the connection to retrieve.</p> |
| Output | <p>Body:</p> <pre>{ "id": "string", "type": "ENDPOINT", "state": "INACTIVE", "callType": "UNKNOWN", "mediaType": "UNKNOWN", "callInfo": "string", "causeCode": 0, "rate": 0, "address": "string", "percentConnected": 0, "incoming": true, "encrypted": true, "authState": "NONE", "mediaCount": 0, "terminals": [{ "rel": "self", "href": "string" }], "startTime": 0, "duration": 0, "parentConfId": "string", "answerable": true, "groupname": "string", "grouptype": "string", "canAddVideo": true, "videoEscalationState": "string", "referredBy": "string" }</pre> |
| Applicable return codes | 200 OK, 404 Unavailable |

conferences/{confID}/connections/{connID}

| | |
|-----------------------------------|---|
| Description | This API hangs up and disconnects a specific connection using <code>connID</code> , on the specified conference call using <code>confID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: DELETE Path: <code>/rest/conferences/<confID>/connections/<connID></code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <code><confID>/connections/<connID></code> <code>confID</code> – The unique identifier of the conference to retrieve. <code>connID</code> – The unique identifier for the connection to retrieve. |
| Output | Body: 0 |
| Applicable return codes | 200 OK, 404 Unavailable, 500 Server Error |

conferences/{confID}/terminals

| | |
|-----------------------------------|---|
| Description | This API provides a list of terminals associated with the specified conference call using <code>confID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: <code>/rest/conferences/<confID>/terminals</code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <code><confID></code> <code>confID</code> – The unique identifier of the conference to retrieve. |
| Output | Body: <pre>[{ "id": "string", "name": "string", "callerID": "string", "address": "string", "systemID": "string", "muted": true, "holdState": "NOTHELD", "termType": "REAL", "muteLocked": true, "canSupportMediaStatus": true, "parentConfId": "string", "parentConnectionId": "string", "fullDescription": true, "mediaServerType": "string", "streamsState": "string" }]</pre> |
| Applicable return codes | 200 OK, 404 Unavailable |

conferences/{confID}/terminals/{termID}

| | |
|-----------------------------------|---|
| Description | This API provides a specific terminal using <code>termID</code> , associated with the specified conference call using <code>confID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: <code>/rest/conferences/<confID>/terminals/<termID></code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <code><confID>/terminals/<termID></code></p> <p><code>confID</code> – The unique identifier of the conference to retrieve.</p> <p><code>termID</code> – The unique identifier for the terminal to retrieve.</p> |
| Output | <p>Body:</p> <pre>{ "id": "string", "name": "string", "callerID": "string", "address": "string", "systemID": "string", "muted": true, "holdState": "NOTHELD", "termType": "REAL", "muteLocked": true, "canSupportMediaStatus": true, "parentConfId": "string", "parentConnectionId": "string", "fullDescription": true, "mediaServerType": "string", "streamsState": "string" }</pre> |
| Applicable return codes | 200 OK, 404 Unavailable |

conferences/{confID}/terminals/{termID}

| | |
|-----------------------------------|---|
| Description | This API hangs up and disconnects a specific terminal using <code>termID</code> , associated with the specified conference call using <code>confID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: DELETE Path: <code>/rest/conferences/<confID>/terminals/<termID></code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <code><confID>/terminals/<termID></code> <code>confID</code> – The unique identifier of the conference to retrieve. <code>termID</code> – The unique identifier for the terminal to retrieve. |
| Output | Body: 0 |
| Applicable return codes | 200 OK, 404 Unavailable, 500 Server Error |

conferences/{confID}/mediastats

| | |
|-----------------------------------|---|
| Description | This API provides a list of connections associated to the specified conference. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/conferences/<confID>/mediastats |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <confID> <confID> – The unique identifier of the conference to retrieve. |

Output

Body:

```
[
  {
    "activeAnnexes": "---",
    "actualBitRate": 64,
    "actualFrameRate": 0,
    "address": "none",
    "bitRate": 64,
    "connectionID": "2",
    "encryptionAlgorithm": "AES-256",
    "encryptionMode": "ctr",
    "encryptionType": "TLS/SDES",
    "errorConcealment": "---",
    "frameRate": 0,
    "index": 0,
    "jitter": 0,
    "latency": 126,
    "maxJitter": 0,
    "mediaAlgorithm": "SirenLPR",
    "mediaDirection": "TX",
    "mediaFormat": "---",
    "mediaState": "OPENED",
    "mediaStream": "audioTx",
    "mediaType": "AUDIO",
    "numberOfErrors": 0,
    "packetLoss": 1,
    "percentPacketLoss": 0.0,
    "prevTermId": "-1",
    "quality": 100,
    "reservationError": "err0",
    "reservationProtocol": "rsvp",
    "reservationState": "reserving",
    "termId": "1",
    "totalPackets": 763,
    "uniqueId": null
  },
  {
```

```
"activeAnnexes": "---",
"actualBitRate": 64,
"actualFrameRate": 0,
"address": "none",
"bitRate": 64,
"connectionID": "2",
"encryptionAlgorithm": "AES-256",
"encryptionMode": "ctr",
"encryptionType": "TLS/SDES",
"errorConcealment": "---",
"frameRate": 0,
"index": 0,
"jitter": 3,
"latency": 0,
"maxJitter": 3,
"mediaAlgorithm": "SirenLPR",
"mediaDirection": "RX",
"mediaFormat": "---",
"mediaState": "OPENED",
"mediaStream": "audioRx",
"mediaType": "AUDIO",
"numberOfErrors": 0,
"packetLoss": 0,
"percentPacketLoss": 0.0,
"prevTermId": "-1",
"quality": 100,
"reservationError": "err0",
"reservationProtocol": "rsvp",
"reservationState": "notReserved",
"termId": "1",
"totalPackets": 759,
"uniqueId": null
},
{
  "activeAnnexes": "---",
  "actualBitRate": 1,
  "actualFrameRate": 0,
  "address": "none",
  "bitRate": 1984,
  "connectionID": "2",
  "encryptionAlgorithm": "AES-256",
  "encryptionMode": "ctr",
  "encryptionType": "TLS/SDES",
  "errorConcealment": "---",
  "frameRate": 0,
  "index": 0,
  "jitter": 0,
  "latency": 152,
  "maxJitter": 0,
  "mediaAlgorithm": "H.264-HP",
  "mediaDirection": "TX",
  "mediaFormat": "1080p",
  "mediaState": "OPENED",
  "mediaStream": "videoTx",
```

```
    "mediaType": "VIDEO",
    "numberOfErrors": 0,
    "packetLoss": 0,
    "percentPacketLoss": 0.0,
    "prevTermId": "-1",
    "quality": 100,
    "reservationError": "err0",
    "reservationProtocol": "rsvp",
    "reservationState": "reserving",
    "termId": "1",
    "totalPackets": 3132,
    "uniqueId": null
  }
{
  "activeAnnexes": "---",
  "actualBitRate": 1535,
  "actualFrameRate": 7,
  "address": "none",
  "bitRate": 1984,
  "connectionID": "2",
  "encryptionAlgorithm": "AES-256",
  "encryptionMode": "ctr",
  "encryptionType": "TLS/SDES",
  "errorConcealment": "LPRv1",
  "frameRate": 7,
  "index": 0,
  "jitter": 0,
  "latency": 152,
  "maxJitter": 16,
  "mediaAlgorithm": "H.264-HP",
  "mediaDirection": "RX",
  "mediaFormat": "1080p",
  "mediaState": "OPENED",
  "mediaStream": "videoRx",
  "mediaType": "VIDEO",
  "numberOfErrors": 0,
  "packetLoss": 19,
  "percentPacketLoss": 0.0,
  "prevTermId": "-1",
  "quality": 100,
  "reservationError": "err0",
  "reservationProtocol": "rsvp",
  "reservationState": "notReserved",
  "termId": "1",
  "totalPackets": 1249,
  "uniqueId": null
}
]
```

| | |
|--------------------------------|-------------------------|
| Applicable return codes | 200 OK, 404 Unavailable |
|--------------------------------|-------------------------|

conferences/{confID}/connections/{connID}/mediastat

| | |
|-----------------------------------|--|
| Description | This API provides media statistics for a specific connection using <code>connID</code> , on the specified conference call using <code>confID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: <code>/rest/conferences/<confID>/connections/<connID>/mediastat</code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <code><confID>/connections/<connID>/mediastat</code></p> <p><code>confID</code> – The unique identifier of the conference to retrieve.</p> <p><code>connID</code> – The unique identifier for the connection to retrieve.</p> |
| Output | <p>Body:</p> <pre>[{ "connectionID": "string", "mediaDirection": "string", "mediaType": "string", "index": 0, "mediaState": "string", "mediaStream": "string", "mediaAlgorithm": "string", "mediaFormat": "string", "encryptionType": "string", "encryptionAlgorithm": "string", "encryptionMode": "string", "errorConcealment": "string", "numberOfErrors": 0, "bitRate": 0, "actualBitRate": 0, "frameRate": 0, "actualFrameRate": 0, "totalPackets": 0, "packetLoss": 0, "percentPacketLoss": 0, "jitter": 0, "maxJitter": 0, "latency": 0, "address": "string", "activeAnnexes": "string", "reservationProtocol": "string", "reservationState": "string", "reservationError": "string", "termId": "string", "prevTermId": "string", "quality": 0, }]</pre> |

```
        "uniqueId": "string"
      }
    ]
```

Applicable return codes 200 OK, 404 Unavailable

conferences/{confID}/mediastats/terminals/{termID}

| | |
|-----------------------------------|--|
| Description | This API provides media statistics for a specific terminal using <code>termID</code> , associated with the specified conference call using <code>confID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: <code>/rest/conferences/<confID>/mediastats/terminals/<termID></code> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <code><confID>/mediastats/terminals/<termID></code></p> <p><code>confID</code> – The unique identifier of the conference to retrieve.</p> <p><code>termID</code> – The unique identifier for the terminal to retrieve.</p> |
| Output | <p>Body:</p> <pre>[{ "connectionID": "string", "mediaDirection": "string", "mediaType": "string", "index": 0, "mediaState": "string", "mediaStream": "string", "mediaAlgorithm": "string", "mediaFormat": "string", "encryptionType": "string", "encryptionAlgorithm": "string", "encryptionMode": "string", "errorConcealment": "string", "numberOfErrors": 0, "bitRate": 0, "actualBitRate": 0, "frameRate": 0, "actualFrameRate": 0, "totalPackets": 0, "packetLoss": 0, "percentPacketLoss": 0, "jitter": 0, "maxJitter": 0, "latency": 0, "address": "string", "activeAnnexes": "string", "reservationProtocol": "string", "reservationState": "string", "reservationError": "string", "termId": "string", "prevTermId": "string", "quality": 0, }]</pre> |

```
        "uniqueId": "string"  
      }  
    ]
```

Applicable return codes 200 OK, 404 Unavailable

conferences/{confID}/termmediastats

| | |
|-----------------------------------|--|
| Description | This API provides quality metrics for all terminal endpoints associated with the specified conference call using <code>confID</code> . |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/conferences/<confID>/termmediastats |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <confID> <code>confID</code> – The unique identifier of the conference to retrieve. |
| Output | Body: [{ "quality": 0, "id": "string", }] |
| Applicable return codes | 200 OK, 404 Unavailable |

Directory

directory/favorites

| | |
|-----------------------------------|---|
| Description | This API provides a list of favorite directory entries, sorted by the favorite's index value. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/directory/favorites |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: ?start=<integer>&limit=<integer></p> <p>Example Query: /rest/directory/favorites?start=<integer>&limit=<integer></p> <p>start=<integer> – An integer representing the starting (1-based) index of the search results. Defaults to 1, if not specified.</p> <p>limit=<integer> – An integer representing the maximum number of search results to return. Defaults to 1, if not specified.</p> |
| Output | <p>Body:</p> <pre>{ "entries": [{ "index": 1, "firstName": "string", "lastName": "string", "displayName": "string", "id": "string", "serverId": "string", "link": { "rel": "self", "href": "string" } }], "next": { "rel": "self", "href": "string" } }</pre> |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

directory/global/server

| | |
|-----------------------------------|--|
| Description | This API provides information about the configured global directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/directory/global/server |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <pre> { "id": "string", "address": "string", "protocol": "LOCAL", Status: "DISABLED", "presenceServer": { "protocol": "NONE", Status: "DISABLED" }, "link": { "rel": "self", "href": "string" } } </pre> |
| Applicable return codes | 200 OK, 500 Server Error |

directory/global/mtddefaultgroupentry

| | |
|-----------------------------------|--|
| Description | This API provides information about the multi-tier directory group entry, if configured. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/directory/global/mtddefaultgroupentry |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <pre> { "id": "string", "serverId": "string", "type": "SINGLE", "displayName": "string", "firstName": "string", "lastName": "string", "fullName": "string", "presence": "OFFLINE", "favorite": true, "readOnly": true, "addedLocally": true, "link": { "rel": "self", "href": "string" } } </pre> |
| Applicable return codes | 200 OK, 404 Unavailable, 500 Server Error |

directory/local/server

| | |
|-----------------------------------|---|
| Description | This API provides information about the configured local directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/directory/local/server |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <pre> { "id": "string", "address": "string", "protocol": "LOCAL", Status: "DISABLED", "presenceServer": { "protocol": "NONE", Status: "DISABLED" }, "link": { "rel": "self", "href": "string" } } </pre> |
| Applicable return codes | 200 OK, 500 Server Error |

directory/local/entries

| | |
|-----------------------------------|--|
| Description | This API adds a new entry to the local directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/directory/local/entries |
| Input and Output Syntax | JSON |
| Access Level | Admin |

Input

Body:

```

{
  "type": <string>,
  "displayName": <string>,
  "firstName": <string>,
  "lastName": <string>,
  "fullName": <string>,
  "email": <string>,
  "homePhone": <string>,
  "workPhone": <string>,
  "mobilePhone": <string>,
  "imageLocation": <string>,
  "favorite": <boolean>,
  "devices": [
    {
      "name": <string>,
      "addressList": [
        {
          "type": <string>,
          "number": <string>,
          "extension": <string>,
          "rate": <string>
        },
        {
          "type": <string>,
          "number": <string>,
          "extension": <string>,
          "rate": <string>
        }
      ]
    },
    {
      "name": <string>,
      "addressList": [
        {
          "type": <string>,
          "number": <string>,
          "extension": <string>,
          "rate": <string>
        },
        {
          "type": <string>,

```

```

        "number": <string>,
        "extension": <string>,
        "rate": <string>
      }
    ]
  }
]
}

```

`type` – The type of directory entry. Valid strings include "SINGLE" and "GROUP".

`displayName` – Display name of the directory entry.

`firstName` – First name of the directory entry.

`lastName` – Last name of the directory entry.

`fullName` – Full name of the directory entry.

`email` – E-mail address of the directory entry.

`homePhone` – Home phone number of the directory entry.

`workPhone` – Work phone number of the directory entry.

`mobilePhone` – Mobile phone number of the directory entry.

`imageLocation` – Location of an image associated with the directory entry. Usually a URL to an avatar, icon, or portrait.

`favorite` – True sets the favorite status. False unsets the favorite status.

`device name` – Name of the directory entry device.

`addressList type` – Type of address for the dialable directory entry device. Valid strings include "IP_ADDRESS", "IP_EXTENSION", "H323_NAME", "SIP", "POTS", "ISDN", "EMAIL", "TELEPHONE", and "MOBILE".

`addressList number` – Number of the directory entry device address.

`addressList extension` – Extension for the number of the directory entry device address.

`addressList rate` – Rate to use when dialing the directory entry device address.

| | |
|--------------------------------|---|
| Output | None |
| Applicable return codes | 201 Created, 400 Bad Request, 405 Not Allowed, 500 Server Error, 507 Insufficient Storage |

directory/local/entries/{entryId}

| | |
|-----------------------------------|--|
| Description | This API provides information about the configured local directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/directory/local/entries/<entryId> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <entryId> entryId - The unique identifier of the entry to retrieve. |
| Output | Body: <pre>{ "id": "string", "serverId": "string", "type": "SINGLE", "displayName": "string", "firstName": "string", "lastName": "string", "fullName": "string", "presence": "OFFLINE", "email": "string", "homePhone": "string", "workPhone": "string", "mobilePhone": "string", "imageLocation": "string", "serverProtocol": "string", "devices": [{ "id": "string", "name": "string", "addressList": [{ "type": "IP_ADDRESS", "number": "string", "extension": "string", "rate": "string" }] }], "favorite": true, "readOnly": true, "addedLocally": true, "link": { "rel": "self", "href": "string" } }</pre> |

}

Applicable return codes 200 OK, 404 Unavailable, 500 Server Error

directory/local/entries/{entryId}

| | |
|-----------------------------------|--|
| Description | This API removes an entry from the local directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: DELETE Path: /rest/directory/local/entries/<entryId> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <entryId> entryId - The unique identifier of the entry to retrieve. |
| Output | None |
| Applicable return codes | 204 No Content, 404 Unavailable, 500 Server Error |

directory/local/entries/{entryId}

| | |
|-----------------------------------|---|
| Description | This API performs an update to an existing entry within the local directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: PUT Path: /rest/directory/local/entries/<entryId> |
| Input and Output Syntax | JSON |
| Access Level | Admin |

Input

The following query parameters are accepted:
<entryId>

entryId – The unique identifier of the entry to retrieve.

Body:

```
{
  "type": <string>,
  "displayName": <string>,
  "firstName": <string>,
  "lastName": <string>,
  "fullName": <string>,
  "email": <string>,
  "homePhone": <string>,
  "workPhone": <string>,
  "mobilePhone": <string>,
  "imageLocation": <string>,
  "favorite": <boolean>,
  "devices": [
    {
      "name": <string>,
      "addressList": [
        {
          "type": <string>,
          "number": <string>,
          "extension": <string>,
          "rate": <string>
        },
        {
          "type": <string>,
          "number": <string>,
          "extension": <string>,
          "rate": <string>
        }
      ]
    },
    {
      "name": <string>,
      "addressList": [
        {
          "type": <string>,
          "number": <string>
```

```

        "extension": <string>,
        "rate": <string>
    },
    {
        "type": <string>,
        "number": <string>,
        "extension": <string>,
        "rate": <string>
    }
]
}
]
}

```

`type` – The type of directory entry. Valid strings include "SINGLE" and "GROUP".

`displayName` – Display name of the directory entry.

`firstName` – First name of the directory entry.

`lastName` – Last name of the directory entry.

`fullName` – Full name of the directory entry.

`email` – E-mail address of the directory entry.

`homePhone` – Home phone number of the directory entry.

`workPhone` – Work phone number of the directory entry.

`mobilePhone` – Mobile phone number of the directory entry.

`imageLocation` – Location of an image associated with the directory entry. Usually a URL to an avatar, icon, or portrait.

`favorite` – True sets the favorite status. False unsets the favorite status.

`device name` – Name of the directory entry device.

`addressList type` – Type of address for the dialable directory entry device. Valid strings include "IP_ADDRESS", "IP_EXTENSION", "H323_NAME", "SIP", "POTS", "ISDN", "EMAIL", "TELEPHONE", and "MOBILE".

`addressList number` – Number of the directory entry device address.

`addressList extension` – Extension for the number of the directory entry device address.

`addressList rate` – Rate to use when dialing the directory entry device address.

| | |
|--------------------------------|--|
| Output | None |
| Applicable return codes | 204 No Content, 400 Bad Request, 404 Unavailable, 500 Server Error |

directory/local/entries/{entryId}

| | |
|-----------------------------------|--|
| Description | This API performs an action on the specified entry within the local directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/directory/local/entries/<entryId> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <entryId></p> <p>entryId – The unique identifier of the entry to retrieve.</p> <p>Body:</p> <pre>{ "action": <string>, "deviceId": <string> }</pre> <p>action – The action to perform on the directory entry. The only valid string is "dial".</p> <p>deviceId – Optional property for dial actions to specify the entry's device ID from which to dial.</p> |
| Output | None |
| Applicable return codes | 204 No Content, 404 Unavailable, 500 Server Error |

directory/local/entries/{entryId}/favorite

| | |
|-----------------------------------|--|
| Description | This API performs an action to set or unset favorite status on an existing entry within the local directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: PUT Path: /rest/directory/local/entries/<entryId>/favorite |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted:</p> <p><entryId></p> <p>entryId – The unique identifier of the entry to retrieve.</p> <p>Body:</p> <pre>{ "favorite": <boolean> }</pre> <p>favorite – True sets the favorite status. False unsets the favorite status.</p> |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 404 Unavailable, 500 Server Error |

directory/local/entries/{entryId}/favorite

| | |
|-----------------------------------|--|
| Description | This API performs an update to the position order of an existing favorite entry within the local directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/directory/local/entries/<entryId>/favorite |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted:</p> <p><entryId></p> <p>entryId – The unique identifier of the entry to retrieve.</p> <p>Body:</p> <pre>{ "index": <integer> }</pre> <p>index – An integer to set the favorite position order index.</p> |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 404 Unavailable, 500 Server Error |

directory/local/entries/{entryId}/members

| | |
|-----------------------------------|---|
| Description | This API performs an action on the members of the specified directory group within the local directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/directory/local/entries/<entryId>/members |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted:</p> <p><entryId></p> <p>entryId – The unique identifier of the entry to retrieve.</p> <p>Body:</p> <pre>{ "action": <string>, "entryId": <string> }</pre> <p>action – The action to perform on the directory group. Valid strings include "query", "add", and "remove".</p> <p>entryId – Optional property for group add/remove actions that may specify an entry ID.</p> |
| Output | None |
| Applicable return codes | 201 Created, 204 No Content, 400 Bad Request, 404 Unavailable, 500 Server Error |

directory/queries

| | |
|-----------------------------------|---|
| Description | This API performs a new asynchronous directory entries search query. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/directory/queries |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "searchString": <string>, "serverScope": <string>, "entryScope": <string>, "groupScope": <string>, "order": <string>, "groupID": <string> }</pre> <p><code>searchString</code> – The value for the query search. If this property is empty in the query then all entries are returned.</p> <p><code>serverScope</code> – The server search scope of the query. Defaults to "ALL" if not specified. Valid strings include "ALL", "LOCAL", "GLOBAL", and "FAVORITES".</p> <p><code>entryScope</code> – The entry type search scope of the query. Defaults to "ANY" if not specified. Valid strings include "SINGLE", "GROUP", "ANY_TIERED", and "ANY".</p> <p><code>groupScope</code> – The group search scope of the query. Defaults to "ONE_LEVEL" if not specified. Valid strings include "ONE_LEVEL" and "RECURSIVE".</p> <p><code>order</code> – The order of the results. Defaults to "DISPLAY_NAME" if not specified. Valid strings include "FIRST_NAME", "LAST_NAME", and "DISPLAY_NAME".</p> <p><code>groupID</code> – A unique identifier of the group.</p> |
| Output | <p>Body:</p> <pre>{ "id": "string" }</pre> |
| Applicable return codes | 201 Created, 500 Server Error |

directory/queries/initial

| | |
|-----------------------------------|--|
| Description | This API provides a new asynchronous directory entries search query, used to retrieve a default list of directory entries. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/directory/queries/initial |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "id": "string" } |
| Applicable return codes | 201 Created, 500 Server Error |

directory/queries/{queryId}

| | |
|-----------------------------------|--|
| Description | This API provides results of a directory entries search for a specified query ID. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/directory/queries/<queryId> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: ?start=<integer>&limit=<integer></p> <p>Example Query: /rest/directory/queries/<queryId>?start=<integer>&limit=<integer></p> <p>queryId – The query ID of the directory search to retrieve.</p> <p>start=<integer> – The starting index of the search query results to return. Defaults to 1 if not specified.</p> <p>limit=<integer> – The maximum number of search query results to return. Defaults to 1 if not specified.</p> |
| Output | <p>Body:</p> <pre>{ "entries": [{ "id": "string", "serverId": "string", "type": "SINGLE", "displayName": "string", "firstName": "string", "lastName": "string", "fullName": "string", "presence": "OFFLINE", "favorite": true, "readOnly": true, "addedLocally": true, "link": { "rel": "self", "href": "string" } }], "next": { "rel": "self", "href": "string" } }</pre> |

| | |
|--------------------------------|---|
| Applicable return codes | 200 OK, 204 No Content, 400 Bad Request, 404 Unavailable, 500 Server Error |
|--------------------------------|---|

directory/queries/{queryId}

| | |
|-----------------------------------|--|
| Description | This API removes a directory entry search query for a specified query ID. |
| Protocol, Method, and Path | Protocol: HTTPS Method: DELETE Path: /rest/directory/queries/<queryId> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <queryId> queryId – The query ID of the directory search to delete. |
| Output | None |
| Applicable return codes | 204 No Content, 404 Unavailable, 500 Server Error |

directory/{serverId}/entries/{entryId}

| | |
|-----------------------------------|---|
| Description | This API provides a specified directory entry from the specified directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/directory/<serverId>/entries/<entryId> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <serverId>/entries/<entryId></p> <p>serverId – The unique identifier of the directory server.</p> <p>entryId – The unique identifier of the entry to retrieve.</p> |
| Output | <p>Body:</p> <pre>{ "id": "string", "serverId": "string", "type": "SINGLE", "displayName": "string", "firstName": "string", "lastName": "string", "fullName": "string", "presence": "OFFLINE", "email": "string", "homePhone": "string", "workPhone": "string", "mobilePhone": "string", "imageLocation": "string", "serverProtocol": "string", "devices": [{ "id": "string", "name": "string", "addressList": [{ "type": "IP_ADDRESS", "number": "string", "extension": "string", "rate": "string" }] }], "favorite": true, "readOnly": true, "addedLocally": true, "link": { "rel": "self",</pre> |

```
    "href": "string"  
  }
```

Applicable return codes 200 OK, 404 Unavailable, 500 Server Error

directory/{serverId}/entries/{entryId}

| | |
|-----------------------------------|--|
| Description | This API performs an action on the specified entry within the specified directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/directory/<serverId>/entries/<entryId> |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <serverId>/entries/<entryId></p> <p>serverId – The unique identifier of the directory server.</p> <p>entryId – The unique identifier of the entry to retrieve.</p> <p>Body:</p> <pre>{ "action": <string>, "deviceId": <string> }</pre> <p>action – The action to perform on the directory entry. The only valid string is "dial".</p> <p>deviceId – Optional property for dial actions to specify the entry's device ID from which to dial.</p> |
| Output | None |
| Applicable return codes | 204 No Content, 400 Bad Request, 404 Unavailable, 500 Server Error, 501 Not Implemented, 507 Insufficient Storage |

directory/{serverId}/entries/{entryId}/favorite

| | |
|-----------------------------------|---|
| Description | This API performs an action to set or unset favorite status on the specified entry within the specified directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: PUT Path: /rest/directory/<serverId>/entries/<entryId>/favorite |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted:</p> <p><serverId>/entries/<entryId>/favorite</p> <p>serverId – The unique identifier of the directory server.</p> <p>entryId – The unique identifier of the entry to retrieve.</p> <p>Body:</p> <pre>{ "favorite": <boolean> }</pre> <p>favorite – True sets the favorite status. False unsets the favorite status.</p> |
| Output | <p>Body:</p> <pre>{ "success": true, "reason": "string" }</pre> |
| Applicable return codes | 200 OK, 400 Bad Request, 404 Unavailable, 500 Server Error |

directory/{serverId}/entries/{entryId}/favorite

| | |
|-----------------------------------|---|
| Description | This API performs an update to the position order of the specified entry within the specified directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/directory/<serverId>/entries/<entryId>/favorite |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted:</p> <p><serverId>/entries/<entryId>/favorite</p> <p>serverId – The unique identifier of the directory server.</p> <p>entryId – The unique identifier of the entry to retrieve.</p> <p>Body:</p> <pre>{ "index": <integer> }</pre> <p>index – An integer to set the favorite position order index.</p> |
| Output | <p>Body:</p> <pre>{ "success": true, "reason": "string" }</pre> |
| Applicable return codes | 200 OK, 400 Bad Request, 404 Unavailable, 500 Server Error |

directory/{serverId}/entries/{entryId}/members

| | |
|-----------------------------------|--|
| Description | This API performs an action on the members of the specified directory group within the specified directory server. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/directory/<serverId>/entries/<entryId>/members |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>The following query parameters are accepted: <serverId>/entries/<entryId>/members</p> <p>serverId – The unique identifier of the directory server.</p> <p>entryId – The unique identifier of the entry to retrieve.</p> <p>Body:</p> <pre>{ "action": <string>, "entryId": <string> }</pre> <p>action – The action to perform on the directory group. Valid strings include "query", "add", and "remove".</p> <p>entryId – Optional property for group add/remove actions that may specify an entry ID.</p> |
| Output | <p>Body:</p> <pre>{ "id": "string" }</pre> |
| Applicable return codes | 201 Created, 204 No Content, 400 Bad Request, 404 Unavailable, 500 Server Error |

directory/import

| | |
|-----------------------------------|--|
| Description | This API performs an import of an XML file containing directory entries. |
| Protocol, Method, and Path | Protocol: HTTPS Method: PUT Path: /rest/directory/import |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | None |
| Applicable return codes | 204 No Content, 400 Bad Request, 500 Server Error |

directory/export

| | |
|-----------------------------------|--|
| Description | This API performs an export of directory entries to an XML file and makes them available for download. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/directory/export |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | None |
| Applicable return codes | 200 OK, 400 Bad Request, 500 Server Error |

Session

session

| | |
|-----------------------------------|--|
| Description | This API creates and authenticates a session on the system. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/session |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "user": <string>, "password": <string> }</pre> <p>user – The user ID for signing into the system.</p> <p>password – The password for signing into the system.</p> |
| Output | <p>Body:</p> <pre>{ "success": true, "loginStatus": { "loginResult": "NOLOCKOUT", "lastLoginClient": "198.168.1.1", "lastLoginTime": 1, "lastLoginClientType": "WEB", "failedLogins": 0, "ispasswordAgeLimitReached": false } "session": { "sessionId": "POWODrZMt/Y4yyMhOo0FXvs5yDdehGAapFhoB", "role": "ADMIN", "clientType": "WEB", "location": "192.168.1.2", "userId": "admin", "creationTime": 1, "isAuthenticated": true, "isStale": false, "isNew": true, "isConnected": false } }</pre> |
| Applicable return codes | 200 SUCCESS, 403 LOG-IN ATTEMPT FAILED |

System

system/controller

| | |
|-----------------------------------|---|
| Description | This API provides the MAC address of the factory-default touch controller that will be automatically paired with this device during out-of-box setup. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/system/controller |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "macAddr": <string> } |
| Applicable return codes | 200 OK |

system/controller

| | |
|-----------------------------------|---|
| Description | This API sets the MAC address of the factory-default touch controller that will be automatically paired with this device during out-of-box setup. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/system/controller |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "macAddr": <string> }</pre> <p>macAddr – The touch controller network MAC address.</p> |
| Output | <p>Body:</p> <pre>{ "success": <boolean> }</pre> |
| Applicable return codes | 200 OK |

system/h323gatekeepers

| | |
|-----------------------------------|--|
| Description | This API provides the status and information of the H323 gatekeeper servers. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/system/h323gatekeepers |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: [{ "state": "ON", "address": "string", "version": "string", "model": "string" }] |
| Applicable return codes | 200 OK, 500 Server Error |

system/logs

| | |
|-----------------------------------|---|
| Description | This API provides the system logs in a .tar file as an octet stream file attachment response. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/system/logs |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | None |
| Applicable return codes | 200 OK, 500 Server Error |

system/apps/environment

| | |
|-----------------------------------|--|
| Description | This API starts the Environment Control app on a paired touch controller. This API only supports one connected TC8 device. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/system/apps/environment |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "result": true } |
| Applicable return codes | 200 OK, 500 Server Error |

system/apps/environment

| | |
|-----------------------------------|---|
| Description | This API hides the Environment Control app on a paired touch controller. This API only supports one connected TC8 device. |
| Protocol, Method, and Path | Protocol: HTTPS Method: DELETE Path: /rest/system/apps/environment |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "result": true } |
| Applicable return codes | 200 OK, 500 Server Error |

system/apps/environment

| | |
|-----------------------------------|--|
| Description | This API gets the status of the Environment Control app on a paired touch controller in the form of a list. This API only supports one connected TC8 device. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/system/apps/environment |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: [{ "macAddress": "00e0db54d243", "environmentControlAppActive": true }] |
| Applicable return codes | 200 OK, 500 Server Error |

system/mode/device

| | |
|-----------------------------------|---|
| Description | This API provides the status of whether the Device Mode application is currently running. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/system/mode/device |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "result": true } |
| Applicable return codes | 200 OK |

system/mode/device

| | |
|-----------------------------------|---|
| Description | This API starts Device Mode if it isn't running. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/system/mode/device |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "success": true, "reason": "string" } |
| Applicable return codes | 200 OK |

system/mode/device

| | |
|-----------------------------------|---|
| Description | This API stops Device Mode if it is running. |
| Protocol, Method, and Path | Protocol: HTTPS Method: DELETE Path: /rest/system/mode/device |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "success": true, "reason": "string" } |
| Applicable return codes | 200 OK |

system/mode/signage

| | |
|-----------------------------------|---|
| Description | This API provides the status of whether the Digital Signage application is currently running. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/system/mode/signage |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "result": true } |
| Applicable return codes | 200 OK |

system/mode/signage

| | |
|-----------------------------------|--|
| Description | This API starts Digital Signage if it isn't running. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/system/mode/signage |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "success": true, "reason": "string" } |
| Applicable return codes | 200 OK |

system/mode/signage

| | |
|-----------------------------------|--|
| Description | This API stops Digital Signage if it is running. |
| Protocol, Method, and Path | Protocol: HTTPS Method: DELETE Path: /rest/system/mode/signage |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "success": true, "reason": "string" } |
| Applicable return codes | 200 OK |

system/network/ping

| | |
|-----------------------------------|---|
| Description | This API performs an ICMP echo ping to the destination network host. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/system/network/ping |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "destination": <string>, }</pre> <p>destination – The IP address of the destination network host.</p> |
| Output | <p>Body:</p> <pre>{ "icmpReachable": true, "responseTime": 0, "h323Reachable": true, "sipReachable": true }</pre> |
| Applicable return codes | 200 OK, 500 Server Error |

system/network/traceroute

| | |
|-----------------------------------|---|
| Description | This API performs network traceroute to the destination network host. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/system/network/traceroute |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "destination": <string>, }</pre> <p>destination – The IP address of the destination network host.</p> |
| Output | <p>Body:</p> <pre>{ "success": true, "totalHops": 0, "hops": [{ "hopNumber": 0, "address": "string", "dnsName": "string", "responseTimes": [0] }] }</pre> |
| Applicable return codes | 200 OK, 500 Server Error |

system/reboot

| | |
|-----------------------------------|--|
| Description | This API performs a reboot of the system. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/system/reboot |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | Body: { "action": "reboot" } |
| Output | None |
| Applicable return codes | 200 OK, 500 Server Error |

system/resetsettings

| | |
|-----------------------------------|--|
| Description | This API performs a reset of configurations and settings of the system. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/system/resetsettings |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "action": "resetsettings", "keepCertificates": <boolean>, "keepCdr": <boolean>, "keepDirectory": <boolean>, "keepLogs": <boolean> }</pre> <p><boolean> – True retains the setting. False resets the setting.</p> |
| Output | None |
| Applicable return codes | 200 OK, 500 Server Error |

system/sipservers

| | |
|-----------------------------------|--|
| Description | This API provides the status and information of the SIP servers. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/system/sipservers |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "state": "ON", "address": "string", "version": "string", "model": "string" } |
| Applicable return codes | 200 OK, 500 Server Error |

system/status

| | |
|-----------------------------------|--|
| Description | This API provides the status and information of important areas of the system. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/system/status |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | <pre> Body: { "langtag": "PRIMARY_NETWORK_HEADING", "name": "system.status.ipnetwork", "stateList": ["up"] }, { "langtag": "SIP_SERVER_HEADING", "name": "system.status.sipserver", "stateList": ["up"] }, { "langtag": "GATEKEEPER_HEADING", "name": "system.status.gatekeeper", "stateList": ["up"] }, { "langtag": "CALENDAR_SERVICE_HEADING", "name": "system.status.calendar", "stateList": ["up"] }, { "langtag": "MICROPHONES80", "name": "system.status.microphones", "stateList": ["up"] }, { "langtag": "REMOTE_CONTROL_HEADING", "name": "system.status.remotecontrol", "stateList": [</pre> |

```
        "up"
      ]
    },
    {
      "langtag": "AUTO_ANSWER_P2P_HEADING",
      "name": "system.status.autoanswerp2p",
      "stateList": [
        "off"
      ]
    },
    {
      "langtag": "PROVISIONING_SERVICE_HEADING",
      "name": "system.status.provisioning",
      "stateList": [
        "off"
      ]
    },
    {
      "langtag": "SYSTEM_STATUS_GLOBALDIRECTORY_SERVER",
      "name": "system.status.globaldirectory",
      "stateList": [
        "off"
      ]
    },
    {
      "langtag": "MICROPHONES_HEADING",
      "name": "system.status.mr.audio",
      "stateList": [
        "all_up"
      ]
    },
    {
      "langtag": "CAMERAS_HEADING",
      "name": "system.status.mr.camera",
      "stateList": [
        "none_up"
      ]
    }
  }
}
```

| | |
|--------------------------------|--------------------------|
| Applicable return codes | 200 OK, 500 Server Error |
|--------------------------------|--------------------------|

system/time

| | |
|-----------------------------------|--|
| Description | This API provides the current time in milliseconds of the system, using the UTC time zone and the time offset for the currently-set time zone. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/system/time |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "time": 1487369530000, "tzOffset": -300 } |
| Applicable return codes | 200 OK, 500 Server Error |

system/time

| | |
|-----------------------------------|---|
| Description | This API performs a set of the current system time using Unix Epoch Time format (milliseconds). |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/system/time |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "time": <long>, "tzOffset": <integer> }</pre> <p><code>time</code> – The current system time using Unix Epoch Time format in milliseconds.</p> <p><code>tzOffset</code> – An integer that represents the system time zone offset from GMT in minutes (+/-). This isn't required for a POST request.</p> |
| Output | None |
| Applicable return codes | 200 OK |

system/time/local

| | |
|-----------------------------------|---|
| Description | This API provides the current time in a single ISO 8601 formatted string value. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/system/time/local |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "time": "2018-08-10T15:00:59.000-0700" } |
| Applicable return codes | 200 OK, 500 Server Error |

SysTracker

mediastats

| | |
|-----------------------------------|---|
| Description | This API provides the media status value(s) associated with all shared content. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/mediastats |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: <pre> { "vars": [{ "bitrate": 0, "clientID": "cast3", "duration": 256, "framerate": 0, "height": 1440, "sourceID": 2, "sourceType": "MIRACAST", "start": 1505416756873, "width": 1088 }] }</pre> |
| Applicable return codes | 200 OK |

Video

video/local/mute

| | |
|-----------------------------------|--|
| Description | This API provides the status of whether the video is hidden. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/video/local/mute |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | None |
| Output | Body: { "result": true } |
| Applicable return codes | 200 OK |

video/local/mute

| | |
|-----------------------------------|---|
| Description | This API performs a hide or unhide action on the local video. |
| Protocol, Method, and Path | Protocol: HTTPS Method: POST Path: /rest/video/local/mute |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | <p>Body:</p> <pre>{ "mute": <boolean> }</pre> <p><boolean> – True hides local video. False unhides local video.</p> |
| Output | <p>Body:</p> <pre>{ "success": true, "reason": "string" }</pre> |
| Applicable return codes | 200 OK |

video/remote/{confID}/mute

| | |
|-----------------------------------|--|
| Description | This API provides the status of whether the video is hidden on a specified conference. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/video/remote/<confID>/mute |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <confID>/mute confID – The conference ID to retrieve video hidden state from. |
| Output | Body: { "result": true } |
| Applicable return codes | 200 OK, 404 Unavailable |

video/remote/{confID}/mutelock

| | |
|-----------------------------------|---|
| Description | This API provides the status of whether the video hidden with a mute lock is applied to a specified conference. |
| Protocol, Method, and Path | Protocol: HTTPS Method: GET Path: /rest/video/remote/<confID>/mutelock |
| Input and Output Syntax | JSON |
| Access Level | Admin |
| Input | The following query parameters are accepted: <confID> confID – The conference ID to retrieve video hidden state from. |
| Output | Body: { "result": true } |
| Applicable return codes | 200 OK, 404 Unavailable |

Diagnostics

This section describes REST API error codes and logging information.

Error Codes

Error Codes and Descriptions

| Error Code | Number Code | Description |
|------------|--------------------------|--|
| Success | 200 OK | API executed successfully |
| | 201 Created | Entry was created |
| | 204 No Content | Content is not ready, content has been removed, or updated |
| Failed | 400 Bad Request | Request not recognized, could not be executed |
| | 403 Forbidden | Forbidden, authentication denied, pin invalid |
| | 404 Unavailable | Operation or source unavailable |
| | 405 Not Allowed | Requested action is not supported |
| | 415 Error | Payload invalid or missing one or more required fields |
| | 500 Server Error | Unexpected error encountered |
| | 501 Not Implemented | No support for performing requested action |
| | 507 Insufficient Storage | Insufficient storage on the server |
| Success | 200 OK | API executed successfully |

Logging

The following table provides examples of the type of logging available at each level.

Log Levels

| Level | Description |
|-------|--------------------------------|
| 0 | Debug |
| 1 | Detailed events |
| 2 | Flow events |
| 3 | High-level flow events |
| 4 | Minor, recoverable events |
| 5 | Major, eventually fatal errors |
| 6 | Immediately fatal events |