



Poly UC Software and Poly Voice Software

REST API Reference Manual

June 2023 | 3725-88273-001A

Contents

Before You Begin	4
Prerequisite Information	4
Audience and Purpose of This Guide	4
Related Poly and Partner Resources	4
Privacy Policy	5
REST API Commands	6
Configuration Options	6
Delivering REST API Commands Over SIP	6
Commands and Structure	7
Management.BluetoothInfo	7
Management.CallStatus	8
Management.CommunicationInfo	8
Management.ConfigReset	9
Management.DeviceInfo (Deprecated)	10
Management.DeviceInfo Version 2	11
Management.DeviceStats	12
Management.ExportConfiguration	13
Management.FactoryReset	14
Management.GetConfig	15
Management.GetTransferType	16
Management.LedStatus	16
Management.LineInfo (Deprecated)	17
Management.LineInfo Version 2	19
Management.LocationInformation	20
Management.MrPairInfo	21
Management.NetworkInfo	21
Management.NetworkStatistics	23
Management.PC_Port_Info	23
Management.PollForStatus	24
Management.Reboot	25
Management.Restart	25
Management.RunningConfig	26
Management.SessionStats	28
Management.SetConfig	29
Management.SetTransferType	30
Management.SimulateHook	31
Management.SimulateKeyEvent	31
Management.SimulateTextInput	33
Management.SimulateTouch	34
Management.SkypeSignIn	35
Management.SkypeSignOut	35
Management.Uixml	36
Management.UpdateConfiguration	37
Management.WebSignIn	37
Management.WebSignInCancel	38
Network.UploadBgCapture	39
WebCallControl.AnswerCall	40
WebCallControl.CallLogs	41
WebCallControl.CallStatus Version 2	42
WebCallControl.Dial	44
WebCallControl.EndCall	45
WebCallControl.GetCurrentPresence	46
WebCallControl.HoldCall	47
WebCallControl.IgnoreCall	48
WebCallControl.MuteCall	48

WebCallControl.RejectCall	49
WebCallControl.ResumeCall	50
WebCallControl.SendDTMF	51
WebCallControl.SipStatus	51
WebCallControl.TransferCall	53
WebCallControl.WarmTransfer	54
Diagnostics	55
Error Codes	55
Logging	55
Support	57

Before You Begin

This reference manual describes the REST API feature for Poly UC Software (UCS) and Poly Voice Software (PVOS).

The REST API for UCS and PVOS provides a convenient, scalable, portable, and reliable API for interacting with Poly VVX, CCX, Trio, and Edge E Series phones, enabling users to execute certain functions and retrieve information.

This manual assumes that the following phone models are using the indicated minimum software versions:

- VVX Series - UCS 5.9
- Trio 8500 and 8800 - UCS 7.2
- CCX Series, Trio C60, Trio 8300, Edge E Series - PVOS 8.0

Note: The Poly CCX 350 Business Media Phone doesn't support the REST API.

Prerequisite Information

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

- The REST API feature is disabled by default.
- The username is `Polycom`, which is case sensitive. Otherwise, the phone returns an HTTP 401 (unauthorized) error.
- You must change the default administrator password before you can use the REST API.
- A POST request must send the content-type as `application/JSON`. If this content-type is not sent by the client, the phone returns an HTTP 400 error.
- The REST API cannot accept more than 20 KB of data. If more than 20 KB of data is sent, the phone returns an HTTP 413 error.

Parallel processing is not allowed. If one API is being processed and another API is received by the phone, the second request will receive an HTTP 403 error or will be queued for later processing.

Audience and Purpose of This Guide

The primary audience for this manual is the person administering the session initiation protocol (SIP) server, provisioning server(s), VoIP network, and Poly UC Software or Poly Voice Software that enables configuration and management of the phone features. This manual is not intended for end users. This manual provides information primarily for mid-level administrators with experience in networking who understand the basics of open SIP networks and VoIP endpoint environments.

Related Poly and Partner Resources

See the following sites for information related to this product.

- [Poly Support](#) is the entry point to online product, service, and solution support information. Find product-specific information such as Knowledge Base articles, Support Videos, Guide & Manuals, and Software Releases on the Products page, download software for desktop and mobile platforms from Downloads & Apps, and access additional services.
- The [Poly Documentation Library](#) provides support documentation for active products, services, and solutions. The documentation displays in responsive HTML5 format so that you can easily 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 a program where resellers, distributors, solutions providers, and unified communications providers deliver high-value business solutions that meet critical customer needs, making it easy for you to communicate face-to-face using the applications and devices you use every day.
- [Poly Services](#) help your business succeed and get the most out of your investment through the benefits of collaboration. Enhance collaboration for your employees by accessing Poly service solutions, including Support Services, Managed Services, Professional Services, and Training Services.
- With [Poly+](#) you get exclusive premium features, insights and management tools necessary to keep employee devices up, running, and ready for action.

- [Poly Lens](#) enables better collaboration for every user in every workspace. It is designed to spotlight the health and efficiency of your spaces and devices by providing actionable insights and simplifying device management.

Privacy Policy

Poly complies with applicable data privacy and protection laws and regulations.

Poly products and services process customer data in a manner consistent with the [Poly Privacy Policy](#). Please direct comments or questions to privacy@poly.com.

REST API Commands

This section provides information on REST API commands.

Configuration Options

This section provides information on configuration parameters.

Configuration Parameters

Attribute	Permitted Values	Default Value	Notes
apps.restapi.enabled	0 or 1	0	
log.level.change.restapi	0 - 6	4	
apps.restapi.sipNotify.enabled	0 or 1	0	Added in UCS 6.4.2 and PVOS 8.0.
apps.restapi.sipNotify.bindRequired	0 or 1	1	Added in UCS 6.4.2 and PVOS 8.0. An optional setting when <code>apps.restapi.sipNotify.enabled</code> is set to 1. A Bind request requires proprietary message formatting used by a limited number of Poly partners. For new development partners interested in using an additional Bind/Unbind request as an additional security layer, please contact your Poly technical account team for more information on co-development. For general usage of the REST API over SIP messaging, this parameter may be set to 0.

Delivering REST API Commands Over SIP

REST commands are most commonly sent to a phone over HTTP(S) however, if configured to do so, Poly devices may accept REST commands embedded in SIP NOTIFY messages.

REST commands over SIP are available starting in UCS 6.4.2 and PVOS 8.0.

Requirements

- 1 Content-type: application/json
- 2 Include the SIP header Event: ACTION-URI
- 3 The NOTIFY body may contain only a single command.

Example

```
NOTIFY sip:3339940893@172.24.166.15 SIP/2.0
Via:SIP/2.0/UDP 172.24.144.139;branch=z9hG4bK15V50-1203-13281774
From:<sip:0891@as.poly.com;user=phone>;tag=1013373203-1328177630
To:"3339940892"<sip:3339940892@as.poly.com>;tag=9C7D0F7F-C6D7ECB4
Call-ID:aa0213505940202121023979765@172.24.144.139
CSeq:516368070 NOTIFY
Contact:<sip:as.poly.com:5060>
Event: ACTION-URI
Max-Forwards: 10
Content-Type: application/JSON
Content-Length:29
```

```
{
  "command-URI": " /api/v1/callctrl/transferCall",
  "data": { "Digits": "" }
}
```

Commands and Structure

Several commands take as an input a REF or REFERENCE_ID.

This reference ID is either the call handle returned in the output data of the `/api/v2/webCallControl/callStatus` command, or it may be a SIP call-ID. SIP call-ID is available starting in UCS 6.4.2 and PVOS 8.0.0.

Management.BluetoothInfo

This API provides Bluetooth information of the device supporting Bluetooth.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: `/api/v1/mgmt/blueTooth/info`

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "data":{
    "Enabled":"Yes",
    "Version":"4.0",
    "Devices":[
      {
        "Type":"Phone",
        "Connected":"Yes",
        "Name":"OnePlus 5T",
        "Profile":"HFP",
        "Signal Strength":"-200"
      }
    ]
  },
  "Status":"2000"
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.CallStatus

This API provides all the information of calls on the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/webCallControl/callStatus

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "Status": "2000",
  "data":
  {
    "CallHandle": "<9577dd30>",
    "Type": "<Incoming >",
    "Protocol": "<SIP >",
    "CallState": "<Connected >",
    "LineID": "<1>",
    "RemotePartyName": "<3339941432>",
    "RemotePartyNumber": "sip:1432@10.240.10.210",
    "DurationInSeconds": "<7>"
  }
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4007, 4400, 4403, 5000, 5500

Management.CommunicationInfo

This API provides information about any ongoing communication on the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/media/communicationInfo

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "data": {
    "CommunicationType": [
      "<RxOnly/TxOnly/RxTx/None/Idle>",
      "<RxOnly/TxOnly/RxTx/None/Idle>"
    ],
    "FarEndMuteState": [
      "<CONFERENCE_LEG_PHONE_NUMBER>",
      ...
    ],
    "PhoneMuteState": "<True/False>"
  },
  "Status": "2000"
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Notes

This API provides the Receive (Rx) and Transmit (Tx) port open information only. For sending and receiving actual packets, use API `mgmt/media/sessionStats`.

Management.ConfigReset

This API resets the configurations.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path:

`/api/v1/mgmt/configReset` (resets cloud, local, web and device settings)

`/api/v1/mgmt/configReset/cloud`

`/api/v1/mgmt/configReset/local`

`/api/v1/mgmt/configReset/web`

`/api/v1/mgmt/configReset/device`

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.DeviceInfo (Deprecated)

This API provides details about the phone's information.

Important: This version is deprecated. See [Management.DeviceInfo Version 2](#) on page 11.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/device/info

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "Status": "2000",
  "data": {
    "ModelNumber": "<MODEL_NUMBER>",
    "FirmwareRelease": "<FIRMWARE_RELEASE>",
    "DeviceType": "HardwareEndPoint",
    "DeviceVendor": "Polycom",
    "UpTimeSinceLastReboot": "<DAYS_HOURS_MINUTES_SECONDS>",
    "IPV4Address": "<IP_ADDRESS>",
    "IPV6Address": "<IP_ADDRESS>",
    "MACAddress": "<MAC_ADDRESS>",
    "AttachedHardware": [
      "Camera": "<true/false>",
      "EM": [
```

```

    {
      "type": "<paper/LCD>",
      "version": "<VERSION>"},
      ...
    ]
  }
}

```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.DeviceInfo Version 2

This API provides general device information.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v2/mgmt/device/info

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```

{
  "data": {
    "ModelNumber": "<PHONE_MODEL>",
    "DeviceVendor": "Polycom",
    "DeviceType": "HardwareEndpoint",
    "MACAddress": "<MAC_ADDRESS>",
    "Firmware": {
      "Application": "<APPLICATION_VERSION>",
      "Updater": "<UPDATER_VERSION>",
      "BootBlock": "<BOOTBLOCK_VERSION>"
    },
    "IPAddress": "<PHONE_IP_ADDRESS>",
    "IPStack": "IPv4 Only/IPv6 Only/(Dual IPv4/IPv6 stack)",
    "PreferredNetwork": "<IPv4/IPv6>",
    "IPv6Address": "<PHONE_IPV6_ADDRESS>",
    "IPv6LinkAddress": "<LINK_LOCAL_IPV6_ADDRESS>",
    "IPv6ULAddress": "<UNIQUE_LOCAL_IPV6_ADDRESS>",
    "UpTime": {
      "Days": "<NUMBER>"
      "Hours": "<NUMBER>",
    }
  }
}

```

```

    "Minutes": "<NUMBER>",
    "Seconds": "<NUMBER>",
  },
  "AttachedHardware": {
    "Camera": "<True/False>"
    "EM": [
      "Type": "<Paper/LCD>"
      "Version": "<EM_VERSION>"
    ]
  },
  "CanApplyShutdownRequest": "<True/False>",
  "IntendToShutdown": "<True/False>",
  "AppState": "<PHONE_STATE>",
  "ReadyToUse": "<True/False>"
},
"Status": "2000"
}

```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.DeviceStats

This API provides details about the phone's CPU and memory usage.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/device/stats

Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```

{
  "data": {
    "Memory": {
      "Used": "<SIZE_IN_BYTES>",
      "Free": "<SIZE_IN_BYTES>",
      "polyapp": {
        "uordblks": "<SIZE_IN_BYTES>",
        "hblkhd": "<SIZE_IN_BYTES>",
        "arena": "<SIZE_IN_BYTES>",
        "fordblks": "<SIZE_IN_BYTES>"
      },
      "Total": "<SIZE_IN_BYTES>",
    }
  }
}

```

```

    "em": {
      "uordblks": "<SIZE_IN_BYTES>",
      "hblkhd": "<SIZE_IN_BYTES>",
      "arena": "<SIZE_IN_BYTES>",
      "fordblks": "<SIZE_IN_BYTES>"
    },
    "ComAS": "<SIZE_IN_BYTES>",
    "Cached": "<SIZE_IN_BYTES>",
    "SReclaim": "<SIZE_IN_BYTES>",
    "brow": {
      "uordblks": "<SIZE_IN_BYTES>",
      "hblkhd": "<SIZE_IN_BYTES>",
      "arena": "<SIZE_IN_BYTES>",
      "fordblks": "<SIZE_IN_BYTES>"
    },
    "pgui": {
      "uordblks": "<SIZE_IN_BYTES>",
      "hblkhd": "<SIZE_IN_BYTES>",
      "arena": "<SIZE_IN_BYTES>",
      "fordblks": "<SIZE_IN_BYTES>"
    },
    "osd": {
      "uordblks": ""<SIZE_IN_BYTES>",
      "hblkhd": ""<SIZE_IN_BYTES>",
      "arena": ""<SIZE_IN_BYTES>",
      "fordblks": ""<SIZE_IN_BYTES>"
    },
    "ec": {
      "uordblks": ""<SIZE_IN_BYTES>",
      "hblkhd": ""<SIZE_IN_BYTES>",
      "arena": ""<SIZE_IN_BYTES>",
      "fordblks": ""<SIZE_IN_BYTES>"
    },
    "dbs": {
      "uordblks": ""<SIZE_IN_BYTES>",
      "hblkhd": ""<SIZE_IN_BYTES>",
      "arena": ""<SIZE_IN_BYTES>",
      "fordblks": ""<SIZE_IN_BYTES>"
    }
  },
  "CPU": {
    "Current": "<CURRENT_CPU_LOAD>",
    "Average": "<AVERAGE_CPU_LOAD>"
  },
  "RAMDiskSize": ""<SIZE_IN_BYTES>"
},
"Status": "2000"
}

```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.ExportConfiguration

REST Server within the phone receives a POST request with the URL in the data, and then exports the requested configuration to the provided URL.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/config/export

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
    "Url": "<Absolute URL>"
    "ConfigType": "<ConfigFiles|Local|Web|Device|All>"
  }
}
```

URL: Mandatory parameter

ConfigType: Mandatory parameter

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.FactoryReset

This API factory-resets the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/factoryReset

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.GetConfig

This API provides running configuration value for given configuration parameters. The maximum is 20 parameters. The maximum content length for a request is 20 KB.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/config/get

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  [
    "<CONFIG_PARAM_NAME_1>",
    "<CONFIG_PARAM_NAME_2>",
    ...
  ]
}
```

At least one parameter has to be provided.

Output

Success Response:

```
{
  "Status": "2000",
  "data":
  {
    "<CONFIG_PARAM_NAME_1>":
    {
      "value": "<VALUE>",
      "source": "<LOCAL/WEB/CONFIG/DEFAULT>"
    },
    "<CONFIG_PARAM_NAME_2>":
    {
      "value": "<VALUE>",
```

```
        "source": "<LOCAL/WEB/CONFIG/DEFAULT/DEVICE>"
      }
      ...
    }
  }
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4000, 4009, 4400, 4403, 5000, 5500

4009: Parameter count exceeded limit of 20 parameters

Management.GetTransferType

This API returns the current transfer type set on the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/transferType/get

Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
"data": {
  "Type": "<Consultative/Blind/Safe/Unknown>"
},
"Status": "2000"
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.LedStatus

This API provides the LED status of the device.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/ledStatus

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "data": {
    "Lines": [
      {
        "LineNumber": "1",
        "State": "off"
      },
      {
        "LineNumber": "2",
        "Color": "green",
        "State": "active"
      }
    ],
    "MWI": {
      "State": "off"
    },
    "Mute": {
      "State": "off"
    },
    "Speaker": {
      "Color": "green",
      "State": "on"
    },
    "Headset": {
      "State": "off"
    }
  },
  "Status": "2000"
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.LineInfo (Deprecated)

This API provides details about the phone line's information.

Important: This version is deprecated. See [Management.LineInfo Version 2](#) on page 19.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/lineInfo

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "Status": "2000",
  "data": [
    {
      "LineNumber": "<LINE_NUMBER>",
      "Protocol": "<SIP/H323>",
      "SIPAddress": "<SIP_ADDRESS>",
      "ProxyAddress": "<PROXY_ADDRESS>",
      "UserID": "<USER_ID>",
      "Label": "<LABEL>",
      "LineType": "<Shared/Private>",
      "RegistrationStatus": "<Registered/Unregistered>",
      "Port": "<PORT>"
    },
    {
      "LineNumber": "<LINE_NUMBER>",
      "Protocol": "<SIP/H323>",
      "SIPAddress": "<SIP_ADDRESS>",
      "ProxyAddress": "<PROXY_ADDRESS>",
      "AutoDiscovery": "<Enabled/Disabled>"
      "UserID": "<USER_ID>",
      "Label": "<LABEL>",
      "LineType": "<Shared/Private>",
      "RegistrationStatus": "<Registered/Unregistered>",
      "Port": "<PORT>"
    },
    ...
    ...
    ...
  ]
}
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.LineInfo Version 2

This API provides details about the phone's line information.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v2/mgmt/lineInfo

Output Syntax

JSON

Access Level

Admin

Input

The following query parameters are accepted:

line=<1>

Example queries:

<PHONE_IP>/api/v2/mgmt/lineInfo?line=1

Output

Success Response:

```
{
  "data": [
    {
      "ID": "<LINE_NUMBER>",
      "Label": "<LABEL>",
      "Username": "<USERNAME>",
      "Assigned": "<True/False>",
      "RegistrationStatus": "<Registered/Unregistered>",
      "Type": "<Private/Shared>",
      "Active": "<True/False>",
      "DoNotDisturb": "<True/False>",
      "CallAppearances": [
        "<CALL_REFERENCE>",
        "<CALL_REFERENCE>",
        ...
      ],
      "CallServers": [
        {
          "ID": "<SERVER_INDEX>",
          "Address": "<SERVER_ADDRESS>",
          "RegistrationStatus": "<Registered/Unregistered>",
          "Port": "<SERVER_PORT>",
          "Expires": "<REGISTRATION_PERIOD>",
          "Transport": "<DNSNaptr/TcpPreferred/TcpOnly/UdpOnly/TLS/Invalid>",
          "Working": "<True/False>"
        }
      ],
      "ForwardingConfig": {
        "Always": "<CONTACT_NUMBER>",
        "DoNotDisturb": "<CONTACT_NUMBER>",
        "Busy": "<CONTACT_NUMBER>",
        "NoAnswer": "<CONTACT_NUMBER>"
      }
    }
  ],
}
```

```
  ],  
  "Status": "2000"  
}
```

Failure Response:

```
{  
  "Status": "<4xxx/5xxx>"  
}
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.LocationInformation

This API provides the location information of the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/location/info

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{  
  "data": {  
    "Country": "<Value>"  
    "StreetGroup": "<Value>"  
    "House Number": "<Value>"  
    "State": "<Value>"  
    "Place Type ": "<Value>"  
    "District": "<Value>"  
    "Add. Info": "<Value>"  
    "Road SubSection": "<Value>"  
    "City": "<Value>"  
    "Neighbourhood Block": "<Value>"  
    "Landmark": "<Value>"  
    "Post Box": "<Value>"  
    "Division": "<Value>"  
    "Street": "<Value>"  
    "Road Section": "<Value>"  
    "Street1": "<Value>"  
    "Street2": "<Value>"  
    "Place Name": "<Value>"  
    "House Num. Suff.": "<Value>"  
    "Zip Code": "<Value>"  
    "Road Pre Modifier": "<Value>"  
    "Building": "<Value>"  
    "Unit": "<Value>"  
  }  
}
```

```
    "Floor": "<Value>"
    "Additional Code": "<Value>"
    "Room": "<Value>"
    "Postal Name": "<Value>"
    "Seat": "<Value>"
    "Road": "<Value>"
    "Road Branch": "<Value>"
    "Road Post Modifier": "<Value>"
  },
  "Status": 2000
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.MrPairInfo

This API provides information on the status of paired Modular Room Devices.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/mrpair/info

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.NetworkInfo

This API provides details about the phone's network information.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/network/info

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "Status": "2000",
  "data": {
    "DefaultGateway": "0.0.0.0",
    "IPV6Address": "2620:0:1aa0:8071:0:84:0:f9ad/64",
    "IPV4Address": "0.0.0.0",
    "DHCPServer": "10.221.10.20",
    "DHCP": "enabled",
    "UpgradeServer": "",
    "DHCPOption60Format": "ASCII String",
    "DHCPBootServerUseOption": "Static",
    "ZTPStatus": "enabled",
    "DHCPBootServerOption": "160",
    "DHCPBootServerOptionType": "String",
    "LLDP": "enabled",
    "LANPortStatus": "inactive",
    "SubnetMask": "255.0.0.0",
    "AlternateDNSServer": "10.250.64.30",
    "DNSServer": "10.221.10.100",
    "DNSDomain": "iic.eng",
    "LANSpeed": "10MB",
    "SNTPAddress": "172.16.1.15",
    "VLANDiscoveryMode": "Disabled",
    "CDPCompability": "enabled",
    "VLANID": "",
    "VLANIDOption": "129",
    "ProvServerAddress": "10.221.25.42",
    "ProvServerUser": "501",
    "ProvServerType": "FTP",
    "Wifi": {
      "Signal Strength": "89%",
      "State": "Connected",
      "Security Mode": "None",
      "SSID": "WAP371-B0A5-24-OPEN",
      "Duration": "0 days, 0 hours, 1 minutes, 40 seconds"
    }
  }
},
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.NetworkStatistics

This API provides the phone's network statistics information.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/network/stats

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response for Path: /api/mgmt/network/stats

```
{
  "Status": "2000",
  "data":
  {
    "UpTime": "<NETWORK_UPTIME>",
    "RxPackets": <NUM_OF_RX_PACKETS>
    "TxPackets": <NUM_OF_TX_PACKETS>
  }
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.PC_Port_Info

This API provides Phone's PC port information.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/pcPort/info

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "data": {
    "PCPortStatus": "Active",
    "PCMacAddress": "54E1AD1F2ECD"
  },
  "Status": "2000"
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4004, 4400, 4403, 5000, 5500

Management.PollForStatus

This API provides the most recent status of the phone.

Protocol, Method, and Path

Protocol: HTTPS

Method: GET

Path: /api/v1/mgmt/pollForStatus

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "Status": "2000",
  "data": {
    {
      "State": "<Idle|Active|Error>",
      // If State = Idle
      "StateData": "<Time of last call 2015-04-07T19:11:07"
      // If State = Active
      "StateData": "<Active call duration>"
    }
  }
}
```



```
    /// If State = Error
    "StateData": "<Any available Error information>"
    /// e.g. "All Phone applications are not ready."
  }
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.Reboot

This API executes a safeReboot on the phone. safeReboot ensures that all calls on the phone are ended before initiating phone reboot.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/safeReboot

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.Restart

This API executes a safeRestart on the phone. safeRestart ensures that all calls on the phone are ended before initiating a phone restart.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/safeRestart

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

```
{ "Status": "2000" }
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.RunningConfig

This API provides information about running a configuration on the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/device/runningConfig

Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "data": {
    "Network": {
      "IPAddress": "<PHONE_IP_ADDRESS>",
      "SubnetMask": "<SUBNET_MASK>",
      "IPGateway": "<IP_GATEWAY>",
      "IPStack": "IPv4 Only/IPv6 Only/(Dual IPv4/IPv6 stack)",
      "IPv6Address": "<PHONE_IPV6_ADDRESS>",
      "IPv6Gateway": "<IPV6_GATEWAY>",
      "IPv6AddressDiscovery": "<DHCP/Static/SLAAC>",
      "IPv6LinkAddress": "<LINK_LOCAL_IPV6_ADDRESS>",
      "IPv6ULAAddress": "<UNIQUE_LOCAL_IPV6_ADDRESS>",
    }
  }
}
```

```

    "PreferredNetwork": "<IPv4/IPv6>",
    "VLAN": "<VLAN_IDENTIFIER>",
    "VLANFiltering": "<Enabled/Disabled>",
    "CDP": "<Enabled/Disabled>",
    "LLDP": "<Enabled/Disabled>",
    "StormFilterPPS": "<STORM_FILTER_PPS_VALUE>",
    "StormFiltering": "<Enabled/Disabled>",
  },
  "DHCP": {
    "Feature": "<Enabled/Disabled>",
    "Option60Format": "<RFC3925 Binary/ASCII String>",
    "BootServerOptionType": "<IP/String>",
    "BootServerUseOption": "<BOOT_SERVER_USE_OPTION>",
    "BootServerOption": "<BOOT_SERVER_OPTION_NUMBER>",
    "OfferTimeout": "<DHCP_OFFER_TIMEOUT>",
    "VLANDiscovery": "<Disabled/Default/Custom>",
    "VLANDiscoveryOption": "<DHCP_OPTION_FOR_VLAN_DISC>",
  },
  "DNS": {
    "Feature": "<Enabled/Disabled>",
    "PrimaryServer": "<PRIMARY_DNS_SERVER>",
    "SecondaryServer": "<SECONDARY_DNS_SERVER>",
    "Domain": "<DNS_DOMAIN>"
  },
  "Provisioning": {
    "Server": "<PROVISIONING_SERVER_ADDRESS>",
    "ServerType": "<FTP/Trivial FTP/FTPS/HTTP/HTTPS>",
    "User": "<PROVISIONING_USERNAME>",
    "FileTxTries": "<MAX_ATTEMPTS_FOR_FILE_TRANSFER>",
    "RetryWait": "<FILE_TRANSFER_RETRY_WAIT_TIME>",
    "MaxServers": "<MAX_NUMBER_OF_IP_TO_USE_FROM_DNS>",
    "NetworkEnv": "Dial-up/(Cable/DSL)/LAN",
    "TagSerialNo": "<Enabled/Disabled>"
  },
  "Syslog": {
    "Server": "<SYSLOG_SERVER_ADDRESS>",
    "Transport": "<None/TCP/UDP/TLS>",
    "RenderLevel": "<SYSLOG_RENDER_LEVEL>",
    "Facility": "<SYSLOG_FACILITY>",
    "PrependMAC": "<Enabled/Disabled>",
  },
  "SNTP": {
    "Server": "<SNTP_SERVER_ADDRESS>",
    "GMTOffsetSeconds": "<GMT_OFFSET_IN_SECONDS>",
    "GMTOffsetHours": "<GMT_OFFSET_IN_HOURS>"
  },
  "TR069": {
    "Feature": "<Enabled/Disabled>",
    "ACSURL": "<ACS_SERVER_URL>",
    "ACSUsername": "<ACS_USERNAME>",
    "CPEUsername": "<CPE_USERNAME>",
    "PeriodicInform": "<Enabled/Disabled>",
    "InformInterval": "<PERIODIC_INFORM_INTERVAL>",
    "UpgradeManagement": "<Enabled/Disabled>"
  }
},
"Status": "2000"
}

```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.SessionStats

This API provides statistics of active media sessions on the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/media/sessionStats

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "data": [
    {
      "Ref": "<SESSION_REFERENCE>",
      "SRTPCall": "<True/False>",
      "H235Call": "<True/False>",
      "H235DHKey": "<H235_SHARED_SECRET_KEY>"
      "Streams": [
        {
          "Ref": "<AUDIO_STREAM_REFERENCE>",
          "Category": "0:Voice",
          "PacketsReceived": "<RECEIVED_PACKETS>",
          "PacketsSent": "<SENT_PACKETS>",
          "OctetsReceived": "<RECEIVED_OCTETS>",
          "OctetsSent": "<SENT_OCTETS>",
          "PacketsExpected": "<EXPECTED_PACKETS>",
          "PacketsLost": "<LOST_RX_PACKETS>",
          "Jitter": "<RX_JITTER>",
          "MaxJitter": "<MAX_JITTER>",
          "Latency": "<LATENCY>",
          "RxPayloadSize": "<RX_PAYLOAD_SIZE>",
          "TxPayloadSize": "<TX_PAYLOAD_SIZE>",
          "RxCCodec": "<RX_CODEC>",
          "TxCodec": "<TX_CODEC>",
          "RxCMOS": "<REMOTE_CONVERSATIONAL_QUALITY_MOS_SCORE>",
          "TxMOS": "<LOCAL_CONVERSATIONAL_QUALITY_MOS_SCORE>",
          "RxCMSLQ": "<REMOTE_LISTENING_QUALITY_MOS_SCORE>",
          "TxMSLQ": "<LOCAL_LISTENING_QUALITY_MOS_SCORE>",
        },
        {
          "Ref": "<VIDEO_STREAM_REFERENCE>",
          "Category": "1:Video",
          "PacketsReceived": "<RECEIVED_PACKETS>",
          "PacketsSent": "<SENT_PACKETS>",
          "OctetsReceived": "<RECEIVED_OCTETS>",
```

```

    "OctetsSent": "<SENT_OCTETS>",
    "PacketsExpected": "<EXPECTED_PACKETS>",
    "PacketsLost": "<LOST_RX_PACKETS>",
    "Jitter": "<RX_JITTER>",
    "MaxJitter": "<MAX_JITTER>",
    "Latency": "<LATENCY>",
    "RxPayloadSize": "v",
    "TxPayloadSize": "v",
    "RxCCodec": "<RX_CODEC>",
    "TxCodec": "<TX_CODEC>",
    "VideoRxFrameRate": "<VIDEO_RX_FRAMERATE>",
    "VideoRxFrameWidth": "<VIDEO_RX_FRAMEWIDTH>",
    "VideoRxFrameHeight": "<VIDEO_RX_FRAMEHEIGHT>",
    "VideoRxFastUpdateReqCnt":
<VIDEO_RX_FAST_UPDATE_REQUEST_COUNT>",
    "VideoRxActBitrateKbps": "<VIDEO_ACTUAL_BITRATE>",
    "VideoTxFrameRate": "<VIDEO_TX_FRAMERATE>",
    "VideoTxFrameWidth": "<VIDEO_TX_FRAMEWIDTH>",
    "VideoTxFrameHeight": "<VIDEO_TX_FRAMEHEIGHT>",
    "VideoTxConfigBitrateKbps": "<VIDEO_CONFIGURED_BITRATE>",
    "VideoTxFastUpdateReqCnt":
<VIDEO_RX_FAST_UPDATE_REQUEST_COUNT>",
    "VideoTxActBitrateKbps": "<VIDEO_RX_FRAMERATE>",
  }
}
},
...
],
"Status": "2000"
}

```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.SetConfig

This API provides the interface to set the configuration allowing a maximum of 20 parameters and up to 20 KB of content length for a given request.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/config/set

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
```

```
"<CONFIG_PARAM_NAME>": "<CONFIG_PARAM_VALUE>",
"<CONFIG_PARAM_NAME>": "<CONFIG_PARAM_VALUE>",
...
...
}
}
```

At least one parameter has to be provided.

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4000, 4001, 4009, 4400, 4403, 5000, 5500

4000: Invalid parameters

4001: Device busy (In case of Lync web ticket generation or registration any setConfig returns this error code)

4009: Parameter count exceeded limit of 20 parameters

Management.SetTransferType

This API sets the transfer type on the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/transferType/set

Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data": {
    "Type": "<Consultative/Blind/Safe>"
  }
}
```

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4000, 4400, 4403, 5000, 5500

Management.SimulateHook

This API allows the user to simulate off/on hook on the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/simulateHook

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
    "Type": "<Up/Down >"
  }
}
```

Type is a mandatory parameter.

Output

```
{ "Status": "2000" }
```

Failure Response:

```
{ "Status": "4000" }
```

Applicable Return Codes

2000, 4000, 4400, 4403, 5000, 5500

Management.SimulateKeyEvent

This API allows the user to simulate a tap event on the phone UI.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/simulateKeyEvent

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
    "Type": "Tap",
    "KeyName": "<Key Label>"
  }
}
```

Accepted values for <Key Label> on Trio C60 and all CCX except for CCX 350:

- MicMute
- VolDown
- VolUp

Accepted values for <Key Label> on Edge E, VVX, and Trio 8300:

- • SoftKey1
- • SoftKey2
- • SoftKey3
- • SoftKey4
- • SoftKey5
- • Line1
- • Line2
- • LineN, where N is the number of lines visible on the phone UI. Max N allowed 12 (EM or Pagination lines not included)
- ArrowDown
- ArrowLeft
- ArrowRight
- ArrowUp
- Select
- Delete
- Dialpad0
- Dialpad1
- Dialpad2
- Dialpad3
- Dialpad4
- Dialpad5
- Dialpad6
- Dialpad7
- Dialpad8
- Dialpad9
- DialpadStar
- DialpadPound
- Handsfree
- Headset
- MicMute
- Transfer
- VolDown

- VolUp
- Menu
- Redial
- Applications

Output

```
{ "Status": "2000" }
```

Failure Response:

```
{ "Status": "4000" }
```

Applicable Return Codes

2000, 4000, 4400, 4403, 5000, 5500

Management.SimulateTextInput

This API allows the user to input text into text fields in the phone UI.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/simulateTextInput

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
    "Value": "<Input text >",
    "ReplaceText": "<true/false>"
  }
}
```

ReplaceText is optional. If set to true, it replaces any existing text in the phone UI's text field with the value provided.

Output

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4000, 4400, 4403, 5000, 5500

Management.SimulateTouch

This API simulates touch events on the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data": {
    "Type": "<Tap/Press/Release/Swipe>",
    "Positions": [
      {
        "X": "<X_CO-ORDINATE>",
        "Y": "<Y_CO-ORDINATE>"
      }
    ],
    "Duration": "<DURATION_IN_MILLISECONDS>",
    "Easing": "<0-40>"
  }
}
```

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4000, 4009, 4400, 4403, 5000, 5500

Notes

- For “Tap/Press/Release” events, only one position object is accepted.
- For “Swipe” event, a minimum of two and a maximum of five events are accepted.
- “Duration” is the time in milliseconds over which the event is simulated. This is an optional parameter.
- “Easing” represents the type of easing curve applied for the simulation. This is an optional parameter. Refer to [enum QEasingCurve::Type](#) for more information on the accepted values.
- This API will not work if the position provided is outside the screen boundary.
- For softkeys involving “More”, the user must take the view-offset (view-offset="0:0" for the first set of keys, view-offset="320:0" for the second set of keys and so on) in UIXML into consideration. To get the actual coordinates of a softkey on the screen, the user should subtract the x value of view-offset from the x value of the softkey coordinate.

Management.SkypeSignIn

This API allows the user to sign in to Skype for Business on the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/skype/signIn

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data": {
    "Address": "<SIGN_IN_ADDRESS>",
    "User": "<USERNAME>",
    "Password": "<PASSWORD>",
    "Domain": "<DOMAIN>",
    "LockCode": "<LOCK_CODE>"
  }
}
```

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4000, 4002, 4003, 4004, 4400, 4403, 5000, 5500

Notes

- Response will be returned only after the sign in operation succeeds or fails or times out (150 seconds).
- For accounts where domain is optional, an empty string must be passed as the value for "Domain".
- "LockCode" is an optional parameter.

Management.SkypeSignOut

This API allows the user to sign out of Skype for Business on the phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/skype/signOut

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{  
  "Status": "<4xxx/5xxx>"  
}
```

Applicable Return Codes

2000, 4003, 4004, 4400, 4403, 5000, 5500

Management.Uixml

This API provides an XML representation of the phone's user interface.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/uixml

Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{  
  "data": "<XML_DATA>",  
  "Status": "2000"  
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Notes

For softkeys involving "More", the user must take the view-offset (view-offset="0:0" for the first set of keys, view-offset="320:0" for the second set of keys and so on) in UIXML into consideration. To get the actual coordinates of a softkey on the screen, user should subtract the x value of view-offset from the x value of softkey coordinate.

Management.UpdateConfiguration

This API allows the user to sync the phone's configuration with the provisioning server.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/updateConfiguration

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "Status": "2000",
  "data": {
    "ProvisioningState": "<Provisioning status code>"
  }
}
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4001, 4400, 4403, 5000, 5500

Management.WebSignIn

This API allows the user to be signed in to Skype for Business via the web.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/webSignIn

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "data": {
    "Url": "http://aka.ms/sphone",
    "Code": "bpd9m42u1"
  },
  "Status": "2000"
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Management.WebSignInCancel

This API cancels a previous attempt to sign in a user to Skype for Business via the web.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v1/mgmt/webSignInCancel

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "data": {
    "result": "Canceled"
  },
  "Status": "2000"
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Network.UploadBgCapture

This API uploads the phone's captured network packets to a specified URL. If no URL is provided, then the API uploads the captured network packets to the default upload URL, which is the provisioning server.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/mgmt/network/uploadBgCapture

Input/Output Syntax

JSON

Access Level

Admin

Pre-Condition

Packet capture configuration must be enabled to allow background packet capturing on the phone. URL must be configured or has to be passed as an argument with the API call.

Input

```
{
  "data":
  {
    "url": "<Absolute URL>"
  }
}
```

URL: optional parameter

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

If the configurations `diags.pcap.enabled`, `diags.pcap.background.enabled` or both are not enabled, then the phone will reply with HTTP error code 404. You can't execute this API within 10 seconds of a previous `uploadBgCapture` command. A 4001 status code will be returned.

Applicable Return Codes

2000, 4000, 4001, 5000

WebCallControl.AnswerCall

This API answers an incoming call.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: `/api/v1/callctrl/answerCall`

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
    "Ref": "<CALL_REFERENCE>"
  }
}
```

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4000, 4003, 4007, 4400, 4403, 5000, 5500

4003: Operation not allowed. For example, the call isn't in an incoming call.

4007: Call doesn't exist.

Notes

Input is optional. The incoming call represented by Ref will be answered only if it's an active call.

<CALL_REFERENCE> is either the call handle returned in the output data of the `/api/v2/webCallControl/callStatus` command, or it may be a SIP call-ID.

WebCallControl.CallLogs

This API provides the phone's call logs.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path:

`/api/v1/mgmt/callLogs`

`/api/v1/mgmt/callLogs/missed`

`/api/v1/mgmt/callLogs/received`

`/api/v1/mgmt/callLogs/placed`

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response for Path: `/api/v1/callctrl/callLogs`:

```
{
  "Status": "2000",
  "data":
  {
    "missed": [
      {
        "LineNumber": "<LINE NUMBER>",
        "StartTime": "<START TIME>",
        "RemotePartyName": "<REMOTE PARTY NAME>",
        "RemotePartyNumber": "<REMOTE PARTY NUMBER>",
        "Duration": "<DURATION>"
      },
      {
        "LineNumber": "<LINE NUMBER>",
        "StartTime": "<START TIME>",
        "RemotePartyName": "<REMOTE PARTY NAME>",
        "RemotePartyNumber": "<REMOTE PARTY NUMBER>",
        "Duration": "<DURATION>"
      },
      ...
      ...
    ],
    "received": [
      {
        "LineNumber": "<LINE NUMBER>",
```

```

        "StartTime": "<START_TIME>",
        "RemotePartyName": "<REMOTE_PARTY_NAME>",
        "RemotePartyNumber": <REMOTE_PARTY_NUMBER>,
        "Duration": "<DURATION>"
    },
    ...
    ...
    ...
],
"received": [
    {
        "LineNumber": "<LINE_NUMBER>",
        "StartTime": "<START_TIME>",
        "RemotePartyName": "<REMOTE_PARTY_NAME>",
        "RemotePartyNumber": <REMOTE_PARTY_NUMBER>,
        "Duration": "<DURATION>"
    },
    {
        "LineNumber": "<LINE_NUMBER>",
        "StartTime": "<START_TIME>",
        "RemotePartyName": "<REMOTE_PARTY_NAME>",
        "RemotePartyNumber": <REMOTE_PARTY_NUMBER>,
        "Duration": "<DURATION>"
    },
    ...
    ...
    ...
],
"placed": [
    {
        "LineNumber": "<LINE_NUMBER>",
        "StartTime": "<START_TIME>",
        "RemotePartyName": "<REMOTE_PARTY_NAME>",
        "RemotePartyNumber": <REMOTE_PARTY_NUMBER>,
        "Duration": "<DURATION>"
    },
    {
        "LineNumber": "<LINE_NUMBER>",
        "StartTime": "<START_TIME>",
        "RemotePartyName": "<REMOTE_PARTY_NAME>",
        "RemotePartyNumber": <REMOTE_PARTY_NUMBER>,
        "Duration": "<DURATION>"
    },
    ...
]
}

```

Failure Response:

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

WebCallControl.CallStatus Version 2

This API provides information about all the calls present on phone.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: /api/v2/webCallControl/callStatus

Input/Output Syntax

JSON

Access Level

Admin

Input

The following query parameters are accepted:

- handle=<9577dd30>
- line=<1>
- sequence=<1>

wAPI accepts only:

- handle or
- line or
- line and sequence

Example queries:

- <PHONE_IP>/api/v2/webCallControl/callStatus
- <PHONE_IP>/api/v2/webCallControl/callStatus?handle=b5576ff8
- <PHONE_IP>/api/v2webCallControl/callStatus?line=1
- <PHONE_IP>/api/v2/webCallControl/callStatus?line=1&sequence=1

Output

Success Response:

```
{
  "Status": "2000",
  "data": [
    {
      "Remote Connection IP": "10.221.88.152",
      "Media Direction": "sendrecv",
      "CallHandle": "b5576ff8",
      "Type": "Incoming",
      "RemotePartyName": "Test 1171",
      "RemotePartyNumber": "1171",
      "CallState": "Connected",
      "Protocol": "SIP",
      "StartTime": "2017-08-24T12:07:49",
      "DurationSeconds": "25",
      "LineID": "1",
      "CallSequence": "1",
      "UIAppearanceIndex": "1*"
      "Ringing": "0",
      "Muted": "0",
      "RTPPort": "2262",
      "RTCPPort": "2263",
    },
    ...
    {
      "CallHandle": "8901078",
      "RemotePartyNumber": "",
      "Type": "Outgoing",
      "UIAppearanceIndex": "0",
      "CallState": "Connected",
      "Ringing": "0",
      "Muted": "0",
      "CallSequence": "1",
      "RemotePartyName": "",
    }
  ]
}
```

```

    "RTCPPort": "2227",
    "Protocol": "PTT",
    "DurationInSeconds": "2",
    "RTPPort": "2226",
    "StartTime": "2017-08-24T11:56:33",
    "LineID": "0",
    "Channel": {
      "Mode": "Ptt",
      "Number": "1",
      "Label": "Channel",
      "State": "WAITING"
    }
  }
]
}

```

Failure Response:

```

{
  "Status": "<4xxx/5xxx>"
}

```

Applicable Return Codes

2000, 4000, 4007, 4400, 4403, 5000, 5500

Notes

During an active call, a "*" will be appended to the UIAppearanceIndex number.

The following case is applicable for Paging/PTT calls:

Channel object will be present only if it's a Paging/PTT call.

WebCallControl.Dial

This API enables a user to initiate a call to a given number. Moreover, this API initiates the call and returns a response as an acknowledgment of request received.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/callctrl/dial

Input/Output Syntax

JSON

Access Level

Admin

Input

```

{
  "data":
  {
    "Dest": "<NUMBER/SIP_URI>",
    "Line": "<LINE_NUMBER>",
    "Type": "<SIP/TEL/H323>"
  }
}

```

Dest – Mandatory parameter

Line – Optional. The default line is Line 1.

Type – Optional. The default type is TEL.

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{  
  "Status": "<4xxx/5xxx>"  
}
```

2000, 4000, 4002, 4400, 4403, 5000, 5500

4002: A line isn't registered.

Notes

The following format is used where two parameters are required. The test@polycom is the SIPURI of the person that sent the invite. Item in blue is the Skype meeting ID.

```
{  
  "data":  
  {  
    "Dest": "test@poly.com;gruu;opaque=app:conf:focus:id:ZTMVDGBH",  
    "Line": "1"  
    "Type": "SIP"  
  }  
}
```

WebCallControl.EndCall

This API ends an active call.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/callctrl/endCall

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{  
  "data":  
  {  
    "Ref": "<CALL_REFERENCE>"  
  }  
}
```

Ref is a mandatory parameter.

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{  
  "Status": "<4xxx/5xxx>"  
}
```

Applicable Return Codes

2000, 4000, 4003, 4007, 4400, 4403, 5000, 5500

Operation is not allowed.

4007: Call doesn't exist.

Notes

Beginning in UCS 6.4.2 software this API ends a call that is on hold.

<CALL_REFERENCE> is either the call handle returned in the output data of the `/api/v2/webCallControl/callStatus` command, or it may be a SIP call-ID.

WebCallControl.GetCurrentPresence

This API provides the phone's presence and is supported only on Skype for Business.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: GET

Path: `/api/v1/mgmt/getPresence`

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{  
  "Status": "2000"  
  "Presence": "<Presence State>"  
}
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4004, 4400, 4403, 5000, 5500

4004: Not supported. This command applies only when the phone is operating in Skype for Business mode.

WebCallControl.HoldCall

This API allows the user to hold an active call.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/callctrl/holdCall

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
    "Ref": "<CALL_REFERENCE>"
  }
}
```

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4000, 4007, 4400, 4403, 5000, 5500

Notes

Input is optional. If input isn't provided, the active call or conference will be put on hold. If the call represented by Ref in input is in a conference, the conference will be put on hold.

<CALL_REFERENCE> is either the call handle returned in the output data of the /api/v2/webCallControl/callStatus command, or it may be a SIP call-ID.

WebCallControl.IgnoreCall

This API allows the user to ignore an incoming call.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/callctrl/ignoreCall

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
    "Ref": "<CALL_REFERENCE>"
  }
}
```

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4003, 4007, 4400, 4403, 5000, 5500

Notes

Input is optional. The call represented by Ref will be ignored only if it's an active call.

<CALL_REFERENCE> is either the call handle returned in the output data of the /api/v2/webCallControl/callStatus command, or it may be a SIP call-ID.

WebCallControl.MuteCall

This API enables a user to mute the phone, if applicable.

Protocol, Method, and Path

Method: POST

Path: /api/v1/callctrl/mute

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
    "state": "<0/1>"
  }
}
```

State is a mandatory parameter.

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4000, 4003, 4007, 4400, 4403, 5000, 5500

4003: The operation isn't allowed. For example, when trying to change the mute state of a call on hold or when an API request is received that doesn't result in a change in mute state.

4007: Call doesn't exist.

WebCallControl.RejectCall

This API allows the user to reject an incoming call.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/callctrl/rejectCall

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
    "Ref": "<CALL_REFERENCE>"
  }
}
```

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{  
  "Status": "<4xxx/5xxx>"  
}
```

Applicable Return Codes

2000, 4003, 4007, 4400, 4403, 5000, 5500

Notes

Input is optional. The call represented by Ref will be rejected only if it's the active call.

WebCallControl.ResumeCall

This API resumes the call that was previously on hold.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/callctrl/resumeCall

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{  
  "data":  
  {  
    "Ref": "<CALL_REFERENCE>"  
  }  
}
```

Ref is a mandatory parameter.

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{  
  "Status": "<4xxx/5xxx>"  
}
```

Applicable Return Codes

2000, 4003, 4007, 4400, 4403, 5000, 5500

Notes

Input is optional. If input isn't provided, the active call resumes if it's on hold.

If the call represented by Ref in input is in a conference, the conference resumes if it's on hold.

<CALL_REFERENCE> is either the call handle returned in the output data of the `/api/v2/webCallControl/callStatus` command, or it may be a SIP call-ID.

WebCallControl.SendDTMF

This API enables a user to send DTMF tones during an active call.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: `/api/v1/callctrl/sendDTMF`

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
    "Digits": "<DIGITS>"
  }
}
```

Digits is a mandatory parameter.

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4000, 4007, 4400, 4403, 5000, 5500

4007: Call doesn't exist.

Notes

The speed of the request execution is improved when the phone receives a consecutive series of SendDTMF API requests that each include just a single digit in the Digits input element.

WebCallControl.SipStatus

This API provides the phone's SIP level details for the user.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/webCallControl/sipStatus

Input/Output Syntax

JSON

Access Level

Admin

Input

None

Output

Success Response:

```
{
  "data":
  {
    "User":
    [
      {
        "Name": "3339940674"
        "TotalEvents": "2"
        "LineNumber": "1"
        "Events":
        [
          {
            "Type": "Register"
            "Overlap": "120"
            "CallID": "b8c9a1cb-cf11d4b-93c1850b@172.24.158.18"
            "RegistrationState": "Registered"
            "Expires": "6965"
          },
          {
            "Type": "BLF"
            "CallID": "7a8900cb-eb6900cb-a65900cb@172.24.158.18"
          }
        ],
        "TotalCalls": "2"
        "Calls":
        [
          {
            "CallState": "Hold"
            "CallID": "f64900cb-e64900cb-9e4900cb@172.24.158.18"
          },
          {
            "CallState": "Connected"
            "CallID": "d64900cb-d64900cb-d64900cb@172.24.158.18"
          }
        ]
      },
      ...
    ],
    "TotalUser": "2"
  }
  "Status": "2000"
}
```

Failure Response:

```
{ "Status": "5000" }
```

Applicable Return Codes

2000, 4400, 4403, 5000, 5500

Notes

The sipStatus of the last call is valid for approximately 40 seconds after the call ends.

WebCallControl.TransferCall

This API enables a user to transfer a call. In addition, this API always executes a blind transfer.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Method: POST

Path: /api/v1/callctrl/transferCall

Input/Output Syntax

JSON

Access Level

Admin

Input

```
{
  "data":
  {
    "Ref": "<CALL REFERENCE>",
    "TransferDest": "<NUMBER>"
  }
}
```

Ref is a mandatory parameter.

TransferDest is a mandatory parameter.

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4000, 4007, 4400, 4403, 5000, 5500

4007: Call doesn't exist.

WebCallControl.WarmTransfer

This API enables a user to transfer an active call to another caller that is currently on hold.

Minimum Supported Version and Usage Limitations

Added in UCS 6.4.2

Added in PVOS 8.0

This API is available for use only when the REST command is sent using SIP. It's described here for completeness but it isn't supported when accessed over HTTP/HTTPS.

Protocol, Method, and Path

Protocol: HTTP/HTTPS

Path: /api/v1/callctrl/warmTransfer

Input/Output Syntax

JSON

Access Level

Admin

```
{
  "data":
  {
    "Ref-A" : "<CALL_REFERENCE for the call to be sent the REFER>",
    "Ref-B" : "<CALL_REFERENCE for the call to be joined with A >"
  }
}
```

Both Ref-A and Ref-B are mandatory.

Output

Success Response:

```
{ "Status": "2000" }
```

Failure Response:

```
{
  "Status": "<4xxx/5xxx>"
}
```

Applicable Return Codes

2000, 4000, 4007, 4400, 4403, 5000, 5500

4007: Call doesn't exist.

Notes

<CALL_REFERENCE> is either the call handle returned in the output data of the /api/v2/webCallControl/callStatus command, or it may be a SIP call-ID.

Diagnostics

This section describes REST API error codes and logging information.

Error Codes

This section describes possible error codes for the UCS/PVOS REST API.

Error Codes and Descriptions

Error Code	Number Code	Description
Success	2000	API executed successfully.
Failed	4000	Invalid input parameters.
	4001	Device busy.
	4002	Line not registered.
	4003	Operation not allowed.
	4004	Operation not supported.
	4005	Line does not exist.
	4006	URLs not configured.
	4007	Call does not exist.
	4008	Configuration export failed.
	4009	Input size limit exceeded.
	4010	Default password not allowed.
	4011	Contact not found
	4400	Bad request
	4403	Forbidden
5000	Failed to process request.	
5500	Internal Server Error	

Logging

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

Log Levels

Log 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

Support

NEED MORE HELP?

poly.com/support

Plantronics, Inc.

345 Encinal Street
Santa Cruz, CA 95060
United States

© 2023 Poly. Bluetooth is a registered trademark of Bluetooth SIG, Inc. DECT is a trademark of ETSI. All trademarks are the property of their respective owners.