Application Programming Interface (API) Reference Guide

Cisco TelePresence SX10
What’s in this guide?

The top menu bar and the entries in the table of contents are all hyperlinks, just click on them to go to the topic.

We recommend you visit our web site regularly for updated versions of the user documentation.

Go to: http://www.cisco.com/go/sx-docs

Table of Contents

Introduction ................................................................. 3
   About this guide ..................................................... 4
   Downloading the user documentation .................... 4
   What’s new in this version ...................................... 5
   New features and improvements ............................. 5
   xConfiguration changes in CE8.0 ......................... 7
   xCommand changes in CE8.0 ................................. 10
   xStatus changes in CE8.0 ....................................... 13

About the API .......................................................... 16
   API fundamentals ................................................. 17
   Connecting to the API .......................................... 17
   Password ............................................................. 17
   Telnet .................................................................... 17
   SSH ...................................................................... 17
   HTTP/HTTPS ....................................................... 18
   API output .......................................................... 19
   The output modes ................................................ 19
   Using the command line ....................................... 20
   API commands ...................................................... 20
   Other commands .................................................. 20
   Command line shortcuts ..................................... 21
   Searching ............................................................ 21
   Value types and formats ...................................... 21
   Commands .......................................................... 22
   Configurations ...................................................... 22
   Status ................................................................. 22
   Events .................................................................... 23
   Feedback mechanism ......................................... 24
   Feedback expressions ......................................... 24
   Terminal connections ......................................... 25

Using HTTP .............................................................. 26
   HTTP XMLAPI Authentication ................................ 26
   HTTP XMLAPI Session Authentication ................ 26
   Getting status and configurations ....................... 27
   Sending commands and configurations ................. 27
   Using HTTP POST ................................................ 27

Feedback from codec over HTTP .......................... 28
   Registering for feedback .................................... 28
   Translating from terminal mode to XML ............... 29

Dos and don'ts .......................................................... 30

xConfiguration commands .................................... 31

xCommand commands ......................................... 72

xStatus commands ................................................ 109

Appendices ............................................................. 152
   About startup scripts ......................................... 153
   The SystemTools commands ............................... 154
   About disconnect cause types ............................ 156
   User documentation on the Cisco web site ........... 157
Chapter 1

Introduction
About this guide

This guide introduces you to the Application Programmer Interface (API) for the Collaboration Endpoint software, and serves as a reference guide for the command line commands for the following product:
- Cisco TelePresence SX10

Downloading the user documentation

You can download the user documentation from the Cisco web site, go to: http://www.cisco.com/go/sx-docs

Guidelines for how to find the documentation on the Cisco web site are included in the User documentation on the Cisco web site appendix.
What's new in this version

This section provides an overview of the new and changed system settings, and the new features and improvements in the Cisco Collaboration Endpoint software version 8.0 (CE8.0) compared to TC7.3.

As CE software is based on TC7, the structure and main functionality remains the same as in TC software.

For more details, we recommend reading the Software release notes:


It is important to consider the upgrade requirements of CE8.0 before upgrading; otherwise upgrading to CE8.0 can leave you with a non-functioning deployment that requires you to downgrade.

New features and improvements

Products
CE8.0 supports the following products:
- MX200 G2
- MX300 G2
- MX700
- MX800
- SX10 Quick Set
- SX20 Quick Set
- SX80
Cisco TelePresence products in EX Series, C Series, and Profile Series are not supported in CE software; use software version TC7.3 or earlier for these products.

User interfaces
Products running CE software, must use the following user interfaces:
- Touch 10 controller, available for all products.
- TRC6 remote control, available for SX10 and SX20.

Touch 8 controller and remote control TRC5 are not supported.

API changes
The number of API commands has been reduced. Some commands have been removed entirely while others are different syntactically to cater for underlying architectural changes.

It is important to consider that current integrations that use the API are likely to need to be reprogrammed in order to work with the new API.

The full list of the changes, as well as a list of new commands are available in this chapter.

Intelligent Proximity for content sharing
Cisco Proximity allows you to automatically pair your device (smartphone, tablet, or laptop) with the video system when the device comes within range. This feature is disabled by default.

Cisco Proximity offers three services: Content sharing to clients, content sharing from clients and basic call control. These services are disabled by default.

The Cisco Proximity clients for smartphones and tablets (Android and iOS), and laptops (Windows and OS X) can be downloaded from ▶ http://proximity.cisco.com. Clients for smartphones and tablets are also available through Google Play (Android) and Apple App Store (iOS).

Microphone LED behaviour
The LED behavior on microphones and Touch 10 has changed. The microphone LED glows and the mute button is active in the following scenarios:
- When initiating an outgoing call and until the call is disconnected.
- When receiving an incoming call and until the call is disconnected.
- When activating the VU meter on the web interface to test the audio levels.

The color indications, green for active and red for muted, have not changed.

PIN code protection
The on-screen Advanced Settings menu can be PIN code protected to prevent unauthorized users from changing the configuration of the video system.
Resolution changes

Collaboration Endpoint Software only supports displays that support 16:9 resolution.

Supports 1080p presentation sharing both locally and in a call, at 5 frames per second.

Remote monitoring

For increased security, it is only possible to take snapshots of the local and far end video streams from the video system's web interface, when a Remote Monitoring option key is installed on the video system.

Remote monitoring is enabled once the option key is added, and the video system is rebooted.

No warning messages or indicators are sent to the users of the video system. Please provide adequate notice to the users that the system administrator may monitor and control the camera and screen.

Removed features

- Cisco CTMS is no longer supported. Other multipoint conferencing solutions (involving Cisco TelePresence Server, Cisco TelePresence MCU, and/or Cisco TelePresence Conductor) may be used instead.
- MediaNet is no longer supported.
xConfiguration changes in CE8.0

New configurations
- Audio Input Microphone [2] Level
- Proximity Mode
- Proximity Services CallControl
- Proximity Services ContentShare FromClients
- Proximity Services ContentShare ToClients
- Video DefaultMainSource

Configurations that are removed
-  means that all configurations starting with  are removed.
  - Audio Microphones Mute Enabled
  - Cameras Camera [n] Flip
  - Conference [1] ActiveControl Mode
  - Conference [1] CallProtocolIPStack
  - Conference [1] FarEndControl SignalCapability
  - Conference [1] Multipoint Mode
  - Conference [1] Presentation *
  - Conference [1] VideoBandwidth *
  - Experimental *
  - FacilityService *
  - Logging Mode
  - Network [1] DHCP RequestTFTPServerAddress
  - Network [1] TrafficControl Mode
  - NetworkServices CTMS Encryption
  - NetworkServices CTMS Mode
  - NetworkServices HTTPS Mode
  - NetworkServices HTTPS OCSP *
  - NetworkServices HTTPS VerifyClientCertificate
  - NetworkServices HTTPS VerifyServerCertificate
  - NetworkServices Mediatet Metadata
  - NetworkServices SSH AllowPublicKey
  - NetworkServices XMLAPI Mode
  - Peripherals Pairing *
  - RTP *
  - Security *
  - SIP ANAT
  - SIP AuthenticateTransferor
  - SIP OCSP *
  - SIP PreferredIPMedia
  - SIP PreferredIPSignaling
  - SIP Profile [1] Line
  - SIP Profile [1] Mailbox
  - SIP Profile [1] Outbound
  - SIP Profile [1] Proxy [n] Discovery
  - SIP Profile [1] TlsVerify
  - SIP Profile [1] Turn BandwidthProbe
  - SIP Profile [1] Turn DiscoverMode
  - SIP Profile [1] Turn DropRflx
  - SIP Profile [1] Type
  - SystemUnit CallLogging Mode
  - SystemUnit MenuLanguage
  - Time OlsonZone
  - UserInterface OSD EncryptionIndicator
UserInterface OSD LanguageSelection
UserInterface TouchPanel DefaultPanel
UserInterface UserPreferences
UserInterface Wallpaper
Video AllowWebSnapshots
Video Input Connector [n] OptimalDefinition Profile
Video Input Connector [n] RGBQuantizationRange
Video Layout DisableDisconnectedLocalOutputs
Video Layout LocalLayoutFamily
Video Layout PresentationDefault View
Video Layout RemoteLayoutFamily
Video Layout ScaleToFrame
Video Layout ScaleToFrameThreshold
Video Layout Scaling
Video OSD EncryptionIndicator
Video OSD LanguageSelection
Video OSD LoginRequired
Video OSD Output
Video Output Connector [n] RGBQuantizationRange
Video PIP *
Video Wallpaper

**Configurations that are modified**

NetworkServices HTTP Mode
OLD: <Off / On>
    Default value: On
NEW: <Off / HTTP+HTTPS / HTTPS>
    Default value: HTTP+HTTPS

Phonebook Server[n] Type
OLD: <VCS / TMS / CUCM>
    Default value: TMS
NEW: <Off / VCS / TMS / CUCM>
    Default value: Off

Standby BootAction
OLD: <None / Preset1 / Preset2 / Preset3 / Preset4 / Preset5 / Preset6 / Preset7 / Preset8 / Preset9 / Preset10 / Preset12 / Preset13 / Preset14 / Preset15 / RestoreCameraPosition / DefaultCameraPosition>
NEW: <None / RestoreCameraPosition / DefaultCameraPosition>

Standby WakeupAction
OLD: <None / Preset1 / Preset2 / Preset3 / Preset4 / Preset5 / Preset6 / Preset7 / Preset8 / Preset9 / Preset10 / Preset12 / Preset13 / Preset14 / Preset15 / RestoreCameraPosition / DefaultCameraPosition>
NEW: <None / RestoreCameraPosition / DefaultCameraPosition>

Time Zone
Change: The list of time zones is updated. The information in the value space is from the tz database, also called the IANA Time Zone Database.

Video Input Connector [n] InputSourceType
OLD: <other / camera / PC / DVD / document_camera / whiteboard>
NEW: <other / camera / PC / mediaplayer / document_camera / whiteboard>

Video Input Connector [n] PresentationSelection
OLD: <Manual / Automatic / OnConnect>
NEW: <Manual / OnConnect>

**Configurations that are renamed**

Audio SoundsAndAlerts KeyTones Mode
Renamed to: UserInterface KeyTones Mode

Cameras Camera [n] Backlight
Renamed to: Cameras Camera [n] Backlight DefaultMode

Cameras Camera [n] Brightness Level
Renamed to: Cameras Camera [n] Brightness DefaultLevel

Conference [1] AutoAnswer Delay
Renamed to: Conference AutoAnswer Delay
Conference [1] AutoAnswer Mode
   Renamed to: Conference AutoAnswer Mode
Conference [1] AutoAnswer Mute
   Renamed to: Conference AutoAnswer Mute
Conference [1] DefaultCall Rate
   Renamed to: Conference DefaultCall Rate
Conference [1] DoNotDisturb DefaultTimeout
   Renamed to: Conference DoNotDisturb DefaultTimeout
Conference [1] Encryption Mode
   Renamed to: Conference Encryption Mode
Conference [1] FarEndControl Mode
   Renamed to: Conference FarEndControl Mode
Conference [1] MaxReceiveCallRate
   Renamed to: Conference MaxReceiveCallRate
Conference [1] MaxTotalReceiveCallRate
   Renamed to: Conference MaxTotalReceiveCallRate
Conference [1] MaxTotalTransmitCallRate
   Renamed to: Conference MaxTotalTransmitCallRate
Conference [1] MaxTransmitCallRate
   Renamed to: Conference MaxTransmitCallRate
NetworkServices NTP Address
   Renamed to: NetworkServices NTP Server [n] Address
SIP Profile [1] Authentication [1] LoginName
   Renamed to: SIP Authentication UserName
   Renamed to: SIP Authentication Password
SIP Profile [1] DefaultTransport
   Renamed to: SIP DefaultTransport
SIP Profile [1] DisplayName
   Renamed to: SIP DisplayName
SIP Profile [1] Ice DefaultCandidate
   Renamed to: SIP Ice DefaultCandidate
SIP Profile [1] Ice Mode
   Renamed to: SIP Ice Mode
SIP Profile [1] Proxy [n] Address
   Renamed to: SIP Proxy [n] Address
SIP Profile [1] Turn Password
   Renamed to: SIP Turn Password
SIP Profile [1] Turn Server
   Renamed to: SIP Turn Server
SIP Profile [1] Turn UserName
   Renamed to: SIP Turn UserName
SIP Profile [1] URI
   Renamed to: SIP URI
SystemUnit ContactInfo Type
   Renamed to: UserInterface ContactInfo Type
Video CamCtrlPip CallSetup Duration
   Renamed to: Video Selfview OnCall Duration
Video CamCtrlPip CallSetup Mode
   Renamed to: Video Selfview OnCall Mode
Video DefaultPresentationSource
   Renamed to: Video Presentation DefaultSource
Video SelfviewDefault FullscreenMode
   Renamed to: Video Selfview Default FullscreenMode
Video SelfviewDefault Mode
   Renamed to: Video Selfview Default Mode
Video SelfviewDefault PIPPosition
   Renamed to: Video Selfview Default PIPPosition
xCommand changes in CE8.0

New commands
- Camera Preset ActivateDefaultPosition
- Camera Preset Show
- Proximity Services Activate
- Proximity Services Deactivate
- SystemUnit OptionKey Remove
- SystemUnit OptionKey RemoveAll
- UserManagement User Passphrase Change
- UserManagement User Passphrase Set

Commands that are removed
- `<path>` * means that all configurations starting with `xCommand <path>` are removed.
- Audio Sound Play
- Audio Sound Stop
- Audio VuMeter *
- Camera PanTiltReset
- Camera PositionActivateFromPreset
- Camera PositionSet
- Camera Preset Snapshot *
- CamCtrlPip
- Experimental *
- FacilityService Dial
- FarEndControl Preset Store
- Key *
- Logging ExtendedLog
- Logging Echo
- Message FarendMessage
- Presentation Byod *
- Provisioning CUCM CAPF OperationStart
- Provisioning CUCM ExtensionMobility *
- Provisioning CUCM ITL Show
- Security FIPSMode Activate
- SystemUnit AdminPassword Set
- SystemUnit ConfigurationProfile *
- UserInterface OSD Close
- UserInterface ScreenShot *
- Video AutoPresentationStart *
- Video Layout SetPresentationView
- Video OSD Close

Commands that are modified
- Call Accept
  - REMOVED: CallType
- Call Disconnect
  - OLD: CallId (r): <0 – 65534>
  - NEW: CallId: <0 – 65534>
- Call Hold
  - OLD: CallId (r): <0 – 65534>
  - NEW: CallId: <0 – 65534>
- Camera PositionReset
  - NEW: Axis: <All/Zoom/Zoom/PanTilt/Zoom/Focus>
- Camera Preset Edit
  - NEW: DefaultPosition: <False/True>
### Camera Preset List
- **NEW:** CameraId: <1 – 7>
- **NEW:** DefaultPosition: <False/True>
- **REMOVED:** PresetId

### Camera Preset Store
- **NEW:** DefaultPosition: <False/True>

### HttpFeedback Deregister
- **OLD:** Required user role: ADMIN
- **NEW:** Required user role: USER

### HttpFeedback Register
- **OLD:** Required user role: ADMIN
- **NEW:** Required user role: USER

### Peripherals List
- **OLD:** Type: <All/BluetoothHeadset/ControlSystem/ISDNLink/Other/TouchPanel>
- **NEW:** Type: <All/ControlSystem/ISDNLink/Other/TouchPanel>

### Presentation Stop
- **NEW:** PresentationSource: <1 – 4>

### User Interface Message Alert Clear
- **OLD:** Required user role: ADMIN
- **NEW:** Required user role: USER

### User Interface Message Prompt Clear
- **OLD:** Required user role: ADMIN
- **NEW:** Required user role: USER

### User Interface Message TextLine Clear
- **OLD:** Required user role: ADMIN
- **NEW:** Required user role: USER

### User Management User Passphrase Set
- **NEW:** YourPassphrase: <S: 0, 255>

### Commands that are renamed or replaced

- **Boot**
  - **Renamed to:** SystemUnit Boot

- **Call DisconnectAll**
  - **Included in:** Call Disconnect

- **Call HoldAll**
  - **Included in:** Call Hold

- **Conference ActiveSpeaker Reset**
  - **Renamed to:** Conference SpeakerLock Release

- **Conference ActiveSpeaker Set**
  - **Renamed to:** Conference SpeakerLock Set

- **DTMF Send**
  - **Renamed to:** Call DTMFSend

- **FarEndControl Camera Move**
  - **Renamed to:** Call FarEndControl Camera Move

- **FarEndControl Camera Stop**
  - **Renamed to:** Call FarEndControl Camera Stop

- **FarEndControl Preset Activate**
  - **Renamed to:** Call FarEndControl RoomPreset Activate

- **FarEndControl Source Select**
  - **Renamed to:** Call FarEndControl Source Select

- **Message **
  - **Renamed to:** User Interface Message *

- **Preset **
  - **Renamed to:** RoomPreset *

- **SystemUnit DateTime Get**
  - **Renamed to:** Time DateTime Get

- **SystemUnit DateTime Set**
  - **Renamed to:** Time DateTime Set

- **OLD:** Year: <2008..2037>
- **NEW:** Year: <2015..2037>
xCommand SystemUnit Diagnostics Run
   Renamed to: xCommand Diagnostics Run

xCommand Video Presentation Set
   Renamed to: xCommand Video PresentationPIP Set

xCommand Video PictureLayoutSet
   Renamed to: xCommand Video Layout LayoutFamily Set

xCommand Video PIP ActiveSpeaker Set
   Renamed to: xCommand Video ActiveSpeakerPIP Set

xCommand Video PIP Presentation Set
   Renamed to: xCommand Video PresentationPIP Set
xStatus changes in CE8.0

New statuses
Call [n] HoldReason
Conference Call [n] Capabilities FarendMessage Mode
Conference Call [n] Capabilities lxChannel Status
H323 Mode Reason
Proximity Services Availability
Standby State
SIP Mailbox URI
SystemUnit Software OptionKeys RemoteMonitoring
UserInterface ContactInfo ContactMethod [n] Number
UserInterface ContactInfo Name

Statuses that are removed

<path> * means that all configurations starting with xStatus <path> are removed.
Call [n] CallPriority
Call [n] ModifyState
Call [n] SecurityStatus
CallDiagnostics [n] * (was Diagnostics Call [n] *)
Cameras Camera [n] DownloadProgress (was Camera [n] DownloadProgress)
Cameras Camera [n] HardwareID (was Camera [n] HardwareID)
Cameras Camera [n] IpAddress (was Camera [n] IpAddress)
Cameras Camera [n] Position * (was Camera [n] Position *)
Cameras Camera [n] UpgradeStatus (was Camera [n] UpgradeStatus)
Conference Call [n] Appearance (was Conference Site [n] Appearance)
Conference Call [n] AttendedTransfer (was Conference Site [n] AttendedTransfer)
Conference Call [n] CalText (was Conference Site [n] CalText)
Conference Call [n] Preserved (was Conference Site [n] Preserved)
Conference Call [n] SecurityStatus (was Conference Site [n] SecurityStatus)
Conference Call [n] UnattendedTransfer (was Conference Site [n] UnattendedTransfer)
Conference Line [n] Appearance [n] *
Conference LoudestSite
Conference Presentation LastLocalSource
Conference Presentation LocalSendingMode
Conference Presentation LocalSource
Conference Presentation Protocol
Conference Presentation Resolution *
Conference SelectedCallProtocol
Conference Site [n] ConferenceExtended
Diagnostics LastRun (was SystemUnit Diagnostics LastRun)
Experimental *
H320 *
ICE *
Logging *
MediaChannels Call [n] *
Network [n] IPv4 DHCP *
Network [n] IPv4 DNS *
Network [n] MTU
Peripherals ConnectProgress [n] *
Peripherals ControllableDevice [n] *
Provisioning CUCM *
Provisioning NextRetry
Provisioning Reason
Provisioning Server
Provisioning Software PreviousUpgrade *
Provisioning Software UpgradeStatus SecondsUntilUpgrade
RoomPreset [n] * (was Preset [n] *)
Security Audit Server Port
SystemUnit Hardware BootSoftware
SystemUnit Hardware MainBoard *
SystemUnit Hardware Module CompatibilityLevel
SystemUnit Hardware Module Identifier
SystemUnit Hardware MonitoringSoftware
SystemUnit Hardware TemperatureThreshold
SystemUnit Hardware UDI
SystemUnit MenuLogo
SystemUnit Software Application
SystemUnit State Subsystem Application
SystemUnit State System
Time Zone Olson
UserInterface OSD Mode
UserInterface OSD Output
Video Input LastConnectedSource
Video Layout *

**Statuses that are modified**

Call [n] Protocol
OLD: String
NEW: <H320/H323/SIP>

Conference Multipoint Mode
OLD: <Auto/CUCMMediaResourceGroupList/MultiSite/MultiWay/Off>
NEW: <Auto/CUCMMediaResourceGroupList/MultiSite/Off>

Diagnostics Message [n] Type
OLD: String
NEW: TemperatureCheck, AudioInternalSpeakerDisabled, and ContactInfoMismatch added to the value space

H323 Mode Status
OLD: String
NEW: <Enabled/Disabled>

Peripherals ConnectedDevice [n] Status
OLD: <Connected/ResponseTimedOut>
NEW: <Connected/ResponseTimedOut/Unpairing/LostConnection>

Peripherals ConnectedDevice [n] Type
OLD: <BluetoothHeadset/Byod/Camera/ControlSystem/ISDNLink/Other/SpeakerTrack Byod/TouchPanel>
NEW: <Byod/Camera/ControlSystem/ISDNLink/Other/SpeakerTrack Byod/TouchPanel>

Peripherals ConnectedDevice [n] UpgradeStatus
OLD: <Downloading/Failed/Installing/None/Succeeded>
NEW: <Downloading/Failed/Installing/InstallationReady/None/Succeeded/Rebooting/Retrying/Aborted/Paused>

Provisioning Status
OLD: <AuthenticationFailed/ConfigError/Failed/Idle/NeedConfig/Provisioned>
NEW: <AuthenticationFailed/ConfigError/Failed/Idle/NeedConfig/Provisioning/Provisioned>

SystemUnit Software OptionKeys Encryption
OLD: String
NEW: <False/True>

SystemUnit Software OptionKeys PremiumResolution
OLD: String
NEW: <False/True>

**Statuses that renamed**

<path> * means that all configurations starting with xStatus <path> are removed.

Camera [n] *
Renamed to: Cameras Camera [n] *

Conference ActiveSpeaker Manual SiteId
Renamed to: Conference SpeakerLock CallId

Conference ActiveSpeaker Mode
Renamed to: Conference SpeakerLock Mode
Conference ActiveSpeaker SiteId
  Renamed to: Conference ActiveSpeaker CallId
Conference Presentation Instance[n] LocalSendingMode
  Renamed to: Conference Presentation LocalInstance[n] SendingMode
Conference Presentation Instance[n] LocalSource
  Renamed to: Conference Presentation LocalInstance[n] Source
Conference Presentation SiteId
  Renamed to: Conference Presentation CallId
Conference Site [n] *
  Renamed to: Conference Call [n] *
Conference Site [n] Hold
  Renamed to: Conference Call [n] Capabilities Hold
NetworkServices NTP Address
  Renamed to: NetworkServices NTP Server [n] Address
SIP Profile [n] *
  Renamed to: SIP *
SIP Profile [n] DirectoryURI Alias [n] URI
  Renamed to: SIP AlternateURI Alias [n] URI
SIP Profile [n] DirectoryURI Primary URI
  Renamed to: SIP AlternateURI Primary URI
SystemUnit ContactInfo
  Renamed to: UserInterface ContactInfo ContactMethod [n] Number
SystemUnit ContactName
  Renamed to: UserInterface ContactInfo Name
SystemUnit Diagnostics Message [n] *
  Renamed to: Diagnostics Message [n] *
SystemUnit Software MaxAudioCalls
  Renamed to: Capabilities Conference MaxAudioCalls
SystemUnit Software MaxVideoCalls
  Renamed to: Capabilities Conference MaxVideoCalls
SystemUnit State MaxNumberOfActiveCalls
  Renamed to: Capabilities Conference MaxActiveCalls
SystemUnit State MaxNumberOfCalls
  Renamed to: Capabilities Conference MaxCalls
Video Input Source [n] Resolution FormatStatus
  Renamed to: Video Input Source [n] FormatStatus
Video Input Source [n] Resolution FormatType
  Renamed to: Video Input Source [n] FormatType
Video PIP ActiveSpeaker Position
  Renamed to: Video ActiveSpeaker PIPPosition
Video PIP Presentation Position
  Renamed to: Video Presentation PIPPosition
Chapter 2

About the API
API fundamentals

This chapter contains a top-level view of the mechanisms supported by the codec API.

Here you can read about how to access the API, how to use the command line and what the different parts of the API can be used for. This chapter also describes how to use the feedback functions that are available for the codec.

The API consists of four major groups:

- Commands
- Configurations
- Status
- Events

These four groups are hierarchically organized, which makes it easier to find related functionality. You can find the complete lists of all commands, configurations and statuses in the following chapters.

Connecting to the API

There are several ways to access the codec API. Regardless of which method you choose, the structure of the API stays the same. Choose the connection method that suits your application best. Before you start, please read this section about the different methods, as some of those may require additional configuration changes before being enabled.

The following configurations, apart from password, can be set from the System configuration menu on the web interface or from the command line interface. All of the examples are for the command line interface.

Password

The video system is delivered with a default user account with full credentials. The user name is admin, and initially, no password is set for the default user.

It is mandatory to set a password for the admin user in order to restrict access to system configuration. You should in addition set a password for any other user with similar credentials.

You can set the password on the codec's web interface. Open a web browser and enter the IP address of the video system in the address bar and log in. Click your user name in the upper right corner and choose Change password in the drop down menu.

Telnet

Telnet can be viewed as the equivalent of the serial protocol in the TCP/IP world. Telnet is disabled by default. Before connecting to the codec with Telnet you have to enable it.

Codec configuration

To enable Telnet service, configure the following setting on the codec. Changing this setting does not require a reboot of the device, but it may take some time to take effect.

xConfiguration NetworkServices Telnet Mode: On

SSH

SSH is a secure TCP/IP connection and it is enabled by default on the codec. It can be disabled. You need a SSH client, such as PuTTY, to access the API over SSH.
HTTP/HTTPS

As HTTP/HTTPS are connectionless protocols, there is no persistent connection. There are several ways of communicating with the API over HTTP.

Codec configuration

In order to enable or disable the HTTP and HTTPS services, configure the following settings on the codec.

- `xConfiguration NetworkServices HTTP Mode`: `<Off/on>
- `xConfiguration NetworkServices HTTPS Mode`: `<Off/on`

Connecting

You can inspect the API by entering the IP address or host name of the codec in your favorite web browser. In the web interface, you can find the API documents under the menu section **System Configuration > API > XML API**. The HTTP POST and GET methods are used to execute commands and get feedback from the codec. This is described in "Using HTTP" on page 26 in this document.
API output

The xPreferences is used to set preferences for the RS-232, Telnet and SSH sessions.

The output modes

- **Terminal**: Line based output for use with line based control systems
- **XML**: XML output for use with control systems that understand XML.
- **JSON**: JSON format is convenient when integrating with web based systems.

The default output mode is **terminal**. To change this you have to define your preferences for each session individually. Examples in this guide are in terminal mode.

To set output mode to XML, issue the command:
```
xPreferences outputmode xml
```

To revert to terminal mode, issue the command:
```
xPreferences outputmode terminal
```

Example: Layout command in terminal mode
```
xCommand Video Layout AssignCall CallId: 2 LayoutId: 1
```

Example: Layout command in XML mode
```
<Command>
  <Video>
    <Layout>
      <AssignCall command="True">
        <CallId>2</CallId>
        <LayoutId>1</LayoutId>
      </AssignCall>
    </Layout>
  </Video>
</Command>
```
Using the command line

Help
To get a list of all supported top level commands you can type ? or help after connecting to the TelePresence System using RS-232, Telnet or SSH (Example 1).

Example 1:

```
?  
- User Commands -
  help  xcommand  xconfiguration  xevent  xfeedback
  xgetxml  xdocument  xpreferences  xtransaction  xstatus
  bye  echo  log  systemtools
OK
```

Bye
Typing the bye command closes the command line interface.

API commands

xConfiguration
Configurations are system settings, such as system name and network settings. These are persistent across boots. Refer to “Configurations” on page 22.

xCommand
Commands instruct the codec to execute actions, such as to dial a number or to search the phone book. Refer to “Commands” on page 22.

xStatus
A status contains the current state of the codec, such as connected calls, the status of the gatekeeper registration, connected inputs and output sources. Refer to “Status” on page 22.

xFeedback
The Feedback commands are used to specify what parts of the configuration and status hierarchies to monitor. Feedback is only issued on the RS-232, Telnet or SSH session for which it is specified. If you are connecting to the codec with multiple sessions, you have to define feedback individually for each session. Refer to “Feedback mechanism” on page 24.

xPreferences
The xPreferences command is used to set preferences for the RS-232, Telnet and SSH sessions. Refer to “API output” on page 19.

Echo <on/off>
If echo is set to On the key inputs are displayed when entering text in a command line interface.

xEvent
the xEvent command returns information on which events are available for feedback. Refer to “Events” on page 23.

xGetxml
The xGetxml request returns an XML document based on the location parameter attached to the request. The elements (or a complete document) matching the expression will be returned. Refer to “Feedback mechanism” on page 24.

Other commands

Systemtools
The systemtools commands are a set of command line tools for administrative control and diagnostics. The commands can be used for advanced troubleshooting together with Cisco technical support. Systemtools are not a part of the programming API. Refer to “The SystemTools commands” on page 154.

Log
The log command is used to enable advanced logs. It is only used for debugging the system.
Command line shortcuts

If your client supports it, there are some timesaving shortcuts you can use:

- Tab-completion to complete the commands and arguments.
- Arrow up and arrow down keys to navigate your command history.
- <CTRL-a>: Jump to the beginning of the line.
- <CTRL-e>: Jump to the end of the line.
- <CTRL-r>: Incremental command history search.
- <CTRL-w>: Erase the current line.

Searching

You can use // to search for elements anywhere in the status or configuration hierarchy (Example 1).

You can also combine multiple //’s (Example 2).

** NOTE: The search shortcuts work well for inspecting the API, but should not be used in applications. We recommend that you always use the complete paths to avoid command ambiguity when upgrading to newer firmware releases.**

Example 1:
List all configurations that include words that starts with OUT and HDMI:

```
xconfiguration //out//hdmi
*c xConfiguration Audio Output HDMI 1 Level: 0
*c xConfiguration Audio Output HDMI 1 Mode: On
*c xConfiguration Audio Output HDMI 2 Level: 0
*c xConfiguration Audio Output HDMI 2 Mode: Off
** end
```

Example 2:
Get the resolution width of all connected sources for both inputs and outputs:

```
xStatus //vid//res//wid
*s Video Input Source 1 Resolution Width: 1920
*s Video Input Source 2 Resolution Width: 0
*s Video Input Source 3 Resolution Width: 0
*s Video Input Source 4 Resolution Width: 0
*s Video Output Connector 1 Resolution Width: 1920
*s Video Output Connector 2 Resolution Width: 1280
*s Video Output Connector 3 Resolution Width: 1280
** end
```

Value types and formats

The system supports the following value types:

- **Integer values:** <x..y>
  Defines the valid range for an integer input. x = min value, y = max value.
- **Literal values:** <X/Y/..Z>
  Defines the possible values for a given configuration.
- **String values:** <S: x, y>
  Defines that the valid input for this configuration is a string with minimum length of x and maximum length of y characters. Strings can have rules that further specify their format and length.

Input values that contain spaces need to be quoted

Any values for configurations and commands containing spaces must be enclosed in quotes. Quotes are not necessary for values without spaces.

**Example:**

Correct: xCommand dial number: “my number contains spaces”
Correct: xCommand dial number: 12345
Incorrect: xCommand dial number: my number contains spaces

Case sensitivity

All commands are case-insensitive. All of the following commands will work.

```
XCOMMAND DIAL NUMBER: foo@bar.org
xcommand dial number: foo@bar.org
xCommand Dial Number: foo@bar.org
```
Commands

Commands instruct the codec to execute actions, such as to dial a number or to search the phone book. All commands start with the prefix `xCommand` followed by a command path.

Writing `xCommand ?` on the command line will list all the top level commands.

To view the complete list of commands and their parameters, write `xCommand ??` on the command line.

Command arguments are key-value pairs.

When issuing a `xCommand`, the command consists of one argument and one required parameter. In this document the command usage is described for all `xCommands`, with both required and optional parameters. The optional parameters are in brackets.

**Example:** `xCommand Dial Number: 123`
- `xCommand` is the command prefix. The command to be executed is `Dial`.
- The example contains one argument, `Number: 123`. Number is the key and 123 is its value. The key/value pair is separated with `:`.

Configurations

Configurations are system settings that are persistent across boots. Like commands, also configurations are structured in a hierarchy.

Writing `xConfiguration ?` on the command line lists all the top level configurations.

Writing `xConfiguration ??` lists all configurations and their value spaces.

Writing `xConfiguration` lists all configurations and their current values. To list out only some of the configurations, write `xConfiguration` followed by one or more parts of the configuration paths.

**Example:** Set the H323 Alias ID

Write in:

```
xConfiguration H323 Profile 1 H323Alias ID: "changed@company.com"
```

**Result:**

```
*x xConfiguration H323 Profile 1 H323Alias ID: "changed@company.com"
**end
```
Events

Event returns information about the events that are available for feedback. This overview presents examples of some of the events that are available on the API.

To get an overview of the supported events type ?, ?? or help after xEvent:
- xEvent ? Lists the top level events
- xEvent ?? List all of the available events
- xEvent help  Lists the top level events

The result for events depends on the state of the codec.

Example 1: Outgoing Call Indication
Outgoing Call Indication is an event reported when an outgoing call is about to be dialled. Returns the CalId the call has been assigned.

```
*e OutgoingCallIndication CallId: x
** end
```

Example 2: Call Disconnect
Call Disconnect is an event reported when a call is disconnected. Returns the CallId of the disconnected call and reason for the call’s disconnection.

```
*e CallDisconnect CallId: x CauseValue: 0
  CauseString: "" CauseType: LocalDisconnect
  OrigCallDirection: "outgoing"
** end
```

Example 3: Call Successful
Call Successful is an event reported when a call is connected successfully, that is when all channels are up and established.

```
*e CallSuccessful CallId: 132 Protocol: "h223"
  Direction: "outgoing" CallRate: 768 RemoteURI: "h223:integratorHQ@company.com" EncryptionIn: "off" EncryptionOut: "off"
** end
```

Example 4: FECC Action request
FECC Action request is an event reported when far end is sending FECC commands.

```
*e FeccActionInd Id: 132 Req: 1 Pan: 1 PanRight: 1 Tilt: 0 TiltUp: 0 Zoom: 0 ZoomIn: 0 Focus: 0 FocusIn: 0 Timeout: 300 VideoSrc: 0 m: 0
** end
```

Example 5: TString message received
TString message received is an event reported when far end has sent a TString message.

```
*e TString CallId: 132 Message: "ee"
** end
```

Example 6: SString message received
SString message received is an event reported when far end has sent a SString message.

```
*e SString String: "ee" Id: 132
** end
```
Feedback mechanism

To build solutions that can reliably keep the state between the application and the codec synchronized, you need to set up a notification system to report the changes in the state of the codec.

The API supports notifications on the following:

- Configuration changes
- Status changes
- Event notifications

These notifications will not be sent unless the user has explicitly told the codec to do so. The user is required to subscribe to all the feedback the application needs. This is done by registering feedback expressions. The way of registering feedback expressions varies according to the connection method used.

When using HTTP, the method of handling feedback differs slightly from what is presented in this section. See “Feedback from codec over HTTP” on page 28 for more information.

**WARNING:** A codec may give very much feedback, especially when calls are connected and disconnected. Therefore, you should only subscribe to the feedback that you need.

Never register for all status feedback by issuing xFeedback register /Status. This may give the control application too much data to handle, which may lead to sluggish or unpredictable behavior.

Feedback expressions

The expression used when registering for feedback is a variant of the XPath language. This language describes a way to select nodes from an XML document. CE software contains three main feedback documents:

<table>
<thead>
<tr>
<th>Document</th>
<th>API command</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td>xStatus</td>
<td>/Status</td>
</tr>
<tr>
<td>Configuration</td>
<td>xConfiguration</td>
<td>/Configuration</td>
</tr>
<tr>
<td>Event</td>
<td>xEvent</td>
<td>/Event</td>
</tr>
</tbody>
</table>

The syntax for feedback registering is:

```
xFeedback register <path>
```

Never register for all status feedback by issuing xFeedback register /Status.

It is safe to register for all configuration changes using xFeedback register /Configuration, as configuration changes will most likely not occur that often.

By going through some examples, we can see how this information can be used to build feedback expressions. A good way to verify the expressions is to point your browser to http://<ip-address>/getxml?location=path or to execute xgetxml <path> from the terminal, and check that the output matches the nodes you want feedback on.

Example 1: Microphones Mute status.

**Terminal query**

```
xStatus Audio Microphones Mute
*s Audio Microphones Mute: Off
** end
```

**Equivalent feedback expression**

```
xFeedback register /Status/Audio/Microphones/Mute
```

Example 2: Name of all video input connectors.

**Terminal query**

```
xConfiguration Video Input Connector Name
* c xConfiguration Video Input Connector 1 Name: "NameA"
* c xConfiguration Video Input Connector 2 Name: "NameB"
* c xConfiguration Video Input Connector 3 Name: "NameC"
* c xConfiguration Video Input Connector 4 Name: "NameD"
* c xConfiguration Video Input Connector 5 Name: "NameE"
** end
```

**Equivalent feedback expression**

```
xFeedback register /Configuration/Video/Input/Connector/Name
```

Example 3: Name of video input connector 3.

**Terminal query**

```
xConfiguration Video Input Connector 3 Name
* c xConfiguration Video Input Connector 3 Name: "NameC"
** end
```

**Equivalent feedback expression**

```
xFeedback register /Configuration/Video/Input/Connector[@item='3']/Name
```
Terminal connections

Managing feedback subscriptions

To register, list and deregister feedback expressions you use the command xFeedback and its corresponding sub commands.

The registered expressions are only valid for the currently active connection. If you open two Telnet sessions and register to get feedback in one session, you do not receive feedback in the other session. This also means that if you disconnect from a session, you have to re-register all expressions after reconnecting.

You can register up to 38 expressions.

Feedback output

The feedback output is exactly the same as you get when querying the system using the xConfiguration and xStatus commands. E.g., if you issue the command xStatus Standby Active on the command line the result is:

*s Standby Active: On
** end

If you have registered for feedback on status changes the feedback you get when the system goes to standby-mode will be exactly the same:

*s Standby Active: On
** end

This means that when you are programming against the device you only need to handle one format.

**Example:** Managing feedback subscriptions

A: Register feedback expressions.

Write in: xFeedback register /Status/Audio
Result: ** end
OK

Write in: xFeedback register /Event/CallDisconnect
Result: ** end
OK

Write in: xFeedback register /Configuration/Video/MainVideoSource
Result: ** end
OK

B: List out currently registered expressions.

Write in: xFeedback list
Result: /Configuration/Video/MainVideoSource
/Event/CallDisconnect
/Status/Audio
** end
OK

C: Deregister feedback expressions.

Write in: xFeedback deregister /Event/CallDisconnect
Result: ** end
OK

Write in: xFeedback deregister /Status/Audio
Result: ** end
OK

D: List the new feedback expressions.

Write in: xFeedback list
Result: /Configuration/Video/MainVideoSource
** end
OK
Using HTTP

The codec supports sending commands and configurations over HTTP and HTTPS. It is also possible to retrieve configurations and statuses this way. This interface exposes the same API as the command line, but in XML format.

HTTP XMLAPI Authentication

Access to the XMLAPI requires the user to authenticate using HTTP Basic Access Authentication as a user with ‘ADMIN’ role. Unauthenticated requests prompt a 401 HTTP response containing a Basic Access Authentication challenge. How to use HTTP Basic Access Authentication varies according to the HTTP library or tool that you are using.

If your application will be issuing multiple commands through the API, we highly recommend that you use Session Authentication (see below). The standard basic authentication does a full re-authentication per request, which may affect the performance of your application.

HTTP XMLAPI Session Authentication

Authenticating with your username and password combination for each API request might introduce too much latency for some use-cases. To mitigate this, the API supports a session-based authentication mechanism.

To open a session, issue a POST to http://<ip-address>/xmlapi/session/begin with Basic Access Authentication. The response sets a SessionId-cookie that can be used with subsequent requests.

Example: Starting a HTTP XMLAPI session

Request:
POST /xmlapi/session/begin HTTP/1.1
Authorization: Basic <Base64 encoded authentication string>

Response:
HTTP/1.1 204 No Content
Server: nginx/1.8.0
Connection: keep-alive
Set-Cookie: SessionId=f08102c8ce5aaf8fba23a7238cc2ef464b990e18bfbb7fb048820c0e28955c54; Path=/; HttpOnly

This session counts toward the systems concurrent sessions limit.

With an open session, provide the SessionId cookie to following requests. Your tool/library may do this automatically.

Example: Using a HTTP XMLAPI session

Request:
GET /configuration.xml HTTP/1.1
Cookie: SessionId=f08102c8ce5aaf8fba23a7238cc2ef464b990e18bfbb7fb048820c0e28955c54

Response:
HTTP/1.1 200 OK
Server: nginx/1.8.0
Content-Type: text/xml; charset=UTF-8
Content-Length: 43549
Connection: keep-alive
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Configuration product="Cisco Codec" version="ce8.2.0" apiVersion="4"> ...
</Configuration>

To close a session after use, issue a POST to http://<ip-address>/xmlapi/session/end with the provided cookie.

Example: Closing a HTTP XMLAPI session

Request:
POST /xmlapi/session/end HTTP/1.1
Cookie: SessionId=f08102c8ce5aaf8fba23a7238cc2ef464b990e18bfbb7fb048820c0e28955c54

Response:
HTTP/1.1 204 No Content
Server: nginx/1.8.0
Connection: keep-alive
Set-Cookie: SessionId=; Max-Age=0; Path=/; HttpOnly

Example: URL cheat sheet

The following table contains the main URLs used when accessing the API over HTTP.

<table>
<thead>
<tr>
<th>Method</th>
<th>URL</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>http://&lt;ip-address&gt;/status.xml</td>
<td>Complete status document</td>
</tr>
<tr>
<td>GET</td>
<td>http://&lt;ip-address&gt;/configuration.xml</td>
<td>Complete configuration document</td>
</tr>
<tr>
<td>GET</td>
<td>http://&lt;ip-address&gt;/command.xml</td>
<td>Complete command document</td>
</tr>
<tr>
<td>GET</td>
<td>http://&lt;ip-address&gt;/valuespace.xml</td>
<td>Complete valuespace document</td>
</tr>
<tr>
<td>GET</td>
<td>http://&lt;ip-address&gt;/getxml?location=&lt;path&gt;</td>
<td>Retrieve document based on a path</td>
</tr>
<tr>
<td>POST</td>
<td>http://&lt;ip-address&gt;/putxml</td>
<td>Configurations and commands in HTTP body</td>
</tr>
</tbody>
</table>
Getting status and configurations

Example 1: Get all status entries on the codec.
http://<ip-address>/getxml?location=/Status

Example 2: Get just the audio statuses of the codec.
http://<ip-address>/getxml?location=/Status/Audio

Example 3: Get all configurations of the codec.
http://<ip-address>/getxml?location=/Configuration

Example 4: Get all video configurations of the codec.
http://<ip-address>/getxml?location=/Configuration/Video

Sending commands and configurations

Using HTTP POST

When sending configurations and commands to the codec, it is important that the HTTP header Content-Type is set to text/xml, i.e., Content-Type: text/xml. The body of the POST should contain the XML content.

Example 1: Changing the system name.

Request
POST /putxml HTTP/1.1
Content-Type: text/xml

<?xml version="1.0"?><Configuration><SystemUnit><Name>newName</Name></SystemUnit></Configuration>

Example 2: Setting the camera position.

Request
POST /putxml HTTP/1.1
Content-Type: text/xml

<?xml version="1.0"?><Command><Camera><PositionSet command="True"><CameraId>1</CameraId><Pan>200</Pan><Tilt>200</Tilt></PositionSet></Camera></Command>

Response
HTTP/1.1 200 OK
Content-Type: text/xml
Content-Length: 91

<?xml version="1.0"?><Command><CameraPositionSetResult item="1" status="OK"/></Command>
Feedback from codec over HTTP

To get notifications from the codec, you need to register HTTP feedback expressions. The codec then uses HTTP POST to send feedback messages to the URL supplied in ServerUrl. This means that you have to have a HTTP server running for your application to receive updates from the codec.

Registering for feedback

The command for registering is `xCommand HttpFeedback Register`. The syntax for this command and its arguments are described in this section.

HttpFeedback Register arguments:

- **FeedbackSlot**: The codec can register up to 4 slots of servers requesting HTTP feedback. Set the registering to one of them. 

- **Note**: Avoid using FeedbackSlot 3 in an environment where Cisco TelePresence Management Suite (TMS) is used as TMS uses this feedback slot to register its expressions.

- **ServerUrl**: The URL that you want the codec to post the HTTP feedback messages to.

- **Expression 1-15**: Register the expressions you want to receive feedback on. See the "Feedback mechanism" on page 24 section for more information about the expression formats.

Example: Registering feedback on configuration changes, disconnect events and call status changes.

```xml
<Command>
  <xCommand>
    <HttpFeedback>
      <Register command="True">
        <FeedbackSlot>1</FeedbackSlot>
        <ServerUrl>http://127.0.0.1/myhttppostscripturl</ServerUrl>
        <Expression item="1">/Configuration</Expression>
        <Expression item="2">/Event/CallDisconnect</Expression>
        <Expression item="3">/Status/Call</Expression>
      </Register>
    </HttpFeedback>
  </xCommand>
</Command>
```

Feedback output

When the codec notifies the registered HTTP server about changes, the body contains the same XML as when polling. There is however one small difference. The root-node contains an `Identification` node with children that specify the codec from which the notification originated. This means that you can handle multiple codecs with a single HTTP server URL.

Example: Audio volume changed.

```xml
<Configuration xmlns="http://www.company.com/XML/CUIL/2.0">
  <Identification>
    <SystemName>integrator</SystemName>
    <MACAddress>00:00:de:ad:be:ef</MACAddress>
    <IPAddress>192.168.1.100</IPAddress>
    <ProductType>Cisco Codec</ProductType>
    <ProductID>Cisco Codec C90</ProductID>
    <SWVersion>TC6.0.0.199465</SWVersion>
    <HWBoard>101401-5 [08]</HWBoard>
    <SerialNumber>PH0000000</SerialNumber>
  </Identification>
  <Audio item="1">
    <Volume item="1">60</Volume>
  </Audio>
</Configuration>
```
Translating from terminal mode to XML

Translating commands
The XML commands maintain the same structure as the terminal commands, but they use a parent-child relationship to describe the hierarchy. You can see this structure in the examples below.

Example 1: Setting up a call.
Terminal
   xCommand Dial Number: "12345" Protocol: H323
XML
   <Command>
     <Dial command="True">
       <Number>12345</Number>
       <Protocol>H323</Protocol>
     </Dial>
   </Command>

Example 2: Assigning video layout to a call.
Terminal
   xCommand Video Layout AssignCall CallId: 2 LayoutId: 1
XML
   <Command>
     <Video>
       <Layout>
         <AssignCall command="True">
           <CallId>2</CallId>
           <LayoutId>1</LayoutId>
         </AssignCall>
       </Layout>
     </Video>
   </Command>

Translating configurations
Translating from xConfiguration to XML is similar to commands, but with the addition of a special attribute item="NN" for specifying the index in arrays.

Example: Configuring the input source type for video input connector 2.
Terminal
   xConfiguration Video Input Connector 2
     InputSourceType: camera
XML
   <Configuration>
     <Video>
       <Input>
         <Connector item="2">
           <InputSourceType>camera</InputSourceType>
         </Connector>
       </Input>
     </Video>
   </Configuration>
Dos and don’ts

Here are some best practices when programming the Cisco SX series API.

AVOID remote control emulation
The use of xCommand UserInterface OSD Key Click and xCommand UserInterface OSD Key Press commands is highly discouraged. The commands are still available in the API, but we recommend the use of direct commands, as this ensures backwards compatibility in your integrations. Program against the codec, not the on-screen-display.

DO use complete commands
You should always use complete commands when programming, i.e. always use xConfiguration Video instead of xconf vid. The shortcuts can be used for searches in the API on the command line, but not for programming. The reason for this is that you might end up with ambiguous code when additional commands are added to the API.

DO NOT subscribe to unnecessary feedback
Subscribing to too much feedback may congest the control application. Although the amount of feedback may seem fine in the current version, the amount of feedback may grow in future releases.
Chapter 3

xConfiguration commands
Description of the xConfiguration commands

In this chapter, you can find a complete list of the xConfiguration commands. The examples show either the default value or an example of a value.

We recommend you visit our web site regularly for updated versions of the manual.

Go to:  http://www.cisco.com/go/sx-docs

Audio configuration ................................................................. 35
  xConfiguration Audio DefaultVolume ..................................... 35
  xConfiguration Audio SoundsAndAlerts RingTone ................... 35
  xConfiguration Audio SoundsAndAlerts RingVolume ............... 35
  xConfiguration Audio Input Microphone [2] Level .................. 35
  xConfiguration Audio Input Microphone [1] Level ................... 35

Cameras configuration ......................................................... 36
  xConfiguration Cameras Camera [1] Backlight DefaultMode ...... 36
  xConfiguration Cameras Camera [1] Brightness Mode ............... 36
  xConfiguration Cameras Camera [1] Brightness DefaultLevel ..... 36
  xConfiguration Cameras Camera [1] Focus Mode .................... 36
  xConfiguration Cameras Camera [1] Mirror ........................ 37
  xConfiguration Cameras Camera [1] Whitebalance Mode ......... 37

Conference configuration ................................................... 38
  xConfiguration Conference AutoAnswer Mode ..................... 38
  xConfiguration Conference AutoAnswer Mute ...................... 38
  xConfiguration Conference AutoAnswer Delay ..................... 38
  xConfiguration Conference DefaultCall Rate ....................... 38
  xConfiguration Conference DoNotDisturb DefaultTimeout ...... 39
  xConfiguration Conference Encryption Mode ...................... 39
  xConfiguration Conference FarEndControl Mode .................. 39
  xConfiguration Conference MaxReceiveCallRate ................... 39

xConfiguration Conference MaxTransmitCallRate .................. 40
xConfiguration Conference MaxTotalReceiveCallRate ............ 40
xConfiguration Conference MaxTotalTransmitCallRate .......... 40

Network configuration ....................................................... 41
  xConfiguration Network [1] DNS Domain Name ..................... 41
  xConfiguration Network [1] DNS Server [1..3] Address ........... 41
  xConfiguration Network [1] IEEE8021X Mode ....................... 41
  xConfiguration Network [1] IEEE8021X UseClientCertificate ..... 42
  xConfiguration Network [1] IEEE8021X Identity ................... 42
  xConfiguration Network [1] IEEE8021X Password .................. 42
  xConfiguration Network [1] IEEE8021X AnonynousIdentity ...... 42
  xConfiguration Network [1] IEEE8021X Eap Md5 .................. 43
  xConfiguration Network [1] IEEE8021X Eap Tls ................... 43
  xConfiguration Network [1] IEEE8021X Eap Tls ................... 43
  xConfiguration Network [1] IEEE8021X Eap Peap ................. 43
  xConfiguration Network [1] IPStack ............................... 44
  xConfiguration Network [1] IPv4 Assignment ...................... 44
  xConfiguration Network [1] IPv4 Address ........................ 44
  xConfiguration Network [1] IPv4 Gateway ........................ 44
  xConfiguration Network [1] IPv4 SubnetMask ..................... 45
  xConfiguration Network [1] IPv6 Assignment ...................... 45
  xConfiguration Network [1] IPv6 Address ........................ 45
<table>
<thead>
<tr>
<th>Configuration</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>xConfiguration Network [1] IPv6 DHCP Options</td>
<td>46</td>
</tr>
<tr>
<td>xConfiguration Network [1] MTU</td>
<td>46</td>
</tr>
<tr>
<td>xConfiguration Network [1] QoS Mode</td>
<td>46</td>
</tr>
<tr>
<td>xConfiguration Network [1] QoS DiffServ Audio</td>
<td>46</td>
</tr>
<tr>
<td>xConfiguration Network [1] QoS DiffServ Video</td>
<td>47</td>
</tr>
<tr>
<td>xConfiguration Network [1] QoS DiffServ Data</td>
<td>47</td>
</tr>
<tr>
<td>xConfiguration Network [1] QoS DiffServ Signalling</td>
<td>47</td>
</tr>
<tr>
<td>xConfiguration Network [1] QoS DiffServ ICMPv6</td>
<td>47</td>
</tr>
<tr>
<td>xConfiguration Network [1] QoS DiffServ NTP</td>
<td>48</td>
</tr>
<tr>
<td>xConfiguration Network [1] Remote Access Allow</td>
<td>48</td>
</tr>
<tr>
<td>xConfiguration Network [1] VLAN Voice Mode</td>
<td>48</td>
</tr>
<tr>
<td>xConfiguration Network [1] VLAN Voice VlanId</td>
<td>48</td>
</tr>
<tr>
<td>Network Services configuration</td>
<td>49</td>
</tr>
<tr>
<td>xConfiguration Network Services CDP Mode</td>
<td>49</td>
</tr>
<tr>
<td>xConfiguration Network Services HTTP Mode</td>
<td>49</td>
</tr>
<tr>
<td>xConfiguration Network Services NTP Mode</td>
<td>49</td>
</tr>
<tr>
<td>xConfiguration Network Services NTP Server [1..3] Address</td>
<td>49</td>
</tr>
<tr>
<td>xConfiguration Network Services SIP Mode</td>
<td>50</td>
</tr>
<tr>
<td>xConfiguration Network Services SNMP Mode</td>
<td>50</td>
</tr>
<tr>
<td>xConfiguration Network Services SNMP Host [1..3] Address</td>
<td>50</td>
</tr>
<tr>
<td>xConfiguration Network Services SNMP Community Name</td>
<td>50</td>
</tr>
<tr>
<td>xConfiguration Network Services SNMP System Contact</td>
<td>51</td>
</tr>
<tr>
<td>xConfiguration Network Services SNMP System Location</td>
<td>51</td>
</tr>
<tr>
<td>xConfiguration Network Services SSH Mode</td>
<td>51</td>
</tr>
<tr>
<td>xConfiguration Network Services Telnet Mode</td>
<td>51</td>
</tr>
<tr>
<td>xConfiguration Network Services Welcome Text</td>
<td>51</td>
</tr>
<tr>
<td>Peripherals configuration</td>
<td>52</td>
</tr>
<tr>
<td>xConfiguration Peripherals Profile Touch Panels</td>
<td>52</td>
</tr>
<tr>
<td>Phonebook configuration</td>
<td>52</td>
</tr>
<tr>
<td>xConfiguration Phonebook Server [1] ID</td>
<td>52</td>
</tr>
<tr>
<td>xConfiguration Phonebook Server [1] Type</td>
<td>52</td>
</tr>
<tr>
<td>xConfiguration Phonebook Server [1] URL</td>
<td>53</td>
</tr>
<tr>
<td>Provisioning configuration</td>
<td>53</td>
</tr>
<tr>
<td>xConfiguration Provisioning Connectivity</td>
<td>53</td>
</tr>
<tr>
<td>xConfiguration Provisioning Mode</td>
<td>54</td>
</tr>
<tr>
<td>xConfiguration Provisioning Login Name</td>
<td>54</td>
</tr>
<tr>
<td>xConfiguration Provisioning Password</td>
<td>54</td>
</tr>
<tr>
<td>xConfiguration Provisioning HTTP Method</td>
<td>55</td>
</tr>
<tr>
<td>xConfiguration Provisioning External Manager Address</td>
<td>55</td>
</tr>
<tr>
<td>xConfiguration Provisioning External Manager Alternate Address</td>
<td>55</td>
</tr>
<tr>
<td>xConfiguration Provisioning External Manager Protocol</td>
<td>55</td>
</tr>
<tr>
<td>xConfiguration Provisioning External Manager Path</td>
<td>56</td>
</tr>
<tr>
<td>xConfiguration Provisioning External Manager Domain</td>
<td>56</td>
</tr>
<tr>
<td>Proximity configuration</td>
<td>56</td>
</tr>
<tr>
<td>xConfiguration Proximity Mode</td>
<td>56</td>
</tr>
<tr>
<td>xConfiguration Proximity Services Call Control</td>
<td>56</td>
</tr>
<tr>
<td>xConfiguration Proximity Services Content Sharing From Clients</td>
<td>57</td>
</tr>
<tr>
<td>xConfiguration Proximity Services Content Sharing To Clients</td>
<td>57</td>
</tr>
<tr>
<td>Serial Port configuration</td>
<td>57</td>
</tr>
<tr>
<td>xConfiguration Serial Port Mode</td>
<td>57</td>
</tr>
<tr>
<td>xConfiguration Serial Port Login Required</td>
<td>57</td>
</tr>
<tr>
<td>SIP configuration</td>
<td>58</td>
</tr>
<tr>
<td>xConfiguration SIP Authentication User Name</td>
<td>58</td>
</tr>
<tr>
<td>xConfiguration SIP Authentication Password</td>
<td>58</td>
</tr>
<tr>
<td>xConfiguration SIP Default Transport</td>
<td>58</td>
</tr>
<tr>
<td>xConfiguration SIP Display Name</td>
<td>58</td>
</tr>
<tr>
<td>xConfiguration SIP Ice Mode</td>
<td>59</td>
</tr>
<tr>
<td>xConfiguration SIP Ice Default Candidate</td>
<td>59</td>
</tr>
<tr>
<td>xConfiguration SIP Listen Port</td>
<td>59</td>
</tr>
<tr>
<td>xConfiguration SIP Proxy [1..4] Address</td>
<td>59</td>
</tr>
<tr>
<td>xConfiguration SIP Turn Server</td>
<td>60</td>
</tr>
<tr>
<td>xConfiguration SIP Turn Username</td>
<td>60</td>
</tr>
<tr>
<td>xConfiguration SIP Turn Password</td>
<td>60</td>
</tr>
<tr>
<td>xConfiguration SIP URI</td>
<td>60</td>
</tr>
</tbody>
</table>
xConfiguration Standby Control .................................................................................. 61
xConfiguration Standby Delay .................................................................................. 61
xConfiguration Standby BootAction .......................................................................... 61
xConfiguration Standby StandbyAction .................................................................... 61
xConfiguration Standby WakeupAction ...................................................................... 61

xConfiguration Video Input Connector [2..3] PresentationSelection ......................... 62
xConfiguration Video Input Connector [2..3] Quality .................................................. 62
xConfiguration Video Input Connector [1..3] Name .................................................... 62
xConfiguration Video Input Connector [1..3] InputSourceType ................................... 62

xConfiguration Video Output Connector [1] Resolution ............................................ 63
xConfiguration Video Output Connector [1] OverscanLevel ........................................ 63

xConfiguration Time Zone ....................................................................................... 63
xConfiguration Time DateFormat ............................................................................... 63
xConfiguration Time TimeFormat ................................................................................ 63

xConfiguration UserInterface Language ................................................................... 65
xConfiguration UserInterface KeyTones Mode .......................................................... 65
xConfiguration UserInterface ContactInfo Type ....................................................... 65
xConfiguration UserInterface OSD Output .................................................................. 65

xConfiguration Video DefaultMainSource .................................................................. 66
xConfiguration Video Input Connector [1..3] CameraControl Mode ......................... 66
xConfiguration Video Input Connector [1..3] CameraControl Camerald ..................... 67
xConfiguration Video Input Connector [1..3] InputSourceType ................................... 67
xConfiguration Video Input Connector [1..3] Name .................................................... 67
xConfiguration Video Input Connector [2..3] Quality .................................................. 67
xConfiguration Video Input Connector [2..3] PresentationSelection ......................... 68
xConfiguration Video Input Connector [1..3] Visibility .............................................. 68
xConfiguration Video Monitors .................................................................................. 68
xConfiguration Video Output Connector [1] CEC Mode ............................................. 69
xConfiguration Video Output Connector [1] OverscanLevel ....................................... 69
xConfiguration Video Output Connector [1] Resolution ............................................ 69
xConfiguration Video Presentation DefaultSource ................................................... 70

xConfiguration Video Selfview Default Mode .......................................................... 70
xConfiguration Video Selfview Default FullscreenMode .......................................... 70
xConfiguration Video Selfview Default PIPPosition .................................................. 71
xConfiguration Video Selfview OnCall Mode ............................................................ 71
xConfiguration Video Selfview OnCall Duration ...................................................... 71
Audio configuration

xConfiguration Audio DefaultVolume
Define the default volume for the speakers. The volume is set to this value when you switch on or restart the video system. Use the Touch controller or remote control to change the volume while the video system is running. You may also use API commands (xCommand Audio Volume) to change the volume while the video system is running, and to reset to default value.

Requires user role: USER
Default value: 50

USAGE:
xConfiguration Audio DefaultVolume: DefaultVolume where
DefaultVolume: Integer (0..100)
Select a value between 1 and 100. This corresponds to the dB range from -34.5 dB to 15 dB, in steps of 0.5 dB). If set to 0 the audio is switched off.

xConfiguration Audio SoundsAndAlerts RingTone
Define which ringtone to use for incoming calls.

Requires user role: USER
Default value: Sunrise

USAGE:
xConfiguration Audio SoundsAndAlerts RingTone: RingTone where
RingTone: Sunrise/Mischief/Ripples/Reflections/Vibes/Delight/Evolve/Playful/Ascent/Calculation/Mellow/Ringer
Select a ringtone from the list.
Cameras configuration

xConfiguration Cameras Camera [1] Backlight DefaultMode
This configuration turns backlight compensation on or off. Backlight compensation is useful when there is much light behind the persons in the room. Without compensation the persons will easily appear very dark to the far end.

Requires user role: USER
Default value: Off

**USAGE:**
xConfiguration Cameras Camera [n] Backlight DefaultMode: DefaultMode
where
  DefaultMode: Off/On
  - Off: Turn off the camera backlight compensation.
  - On: Turn on the camera backlight compensation.

xConfiguration Cameras Camera [1] Brightness Mode
Define the camera brightness mode.

Requires user role: USER
Default value: Auto

**USAGE:**
xConfiguration Cameras Camera [n] Brightness Mode: Mode
where
  - Mode: Auto/Manual
    - Auto: The camera brightness is automatically set by the system.

xConfiguration Cameras Camera [1] Brightness DefaultLevel
Define the brightness level. Requires the Cameras Camera [n] Brightness Mode to be set to Manual.

Requires user role: USER
Default value: 20

**USAGE:**
xConfiguration Cameras Camera [n] Brightness DefaultLevel: DefaultLevel
where
  DefaultLevel: Integer (1..31)
  - The brightness level.

xConfiguration Cameras Camera [1] Focus Mode
Define the camera focus mode.

Requires user role: USER
Default value: Auto

**USAGE:**
xConfiguration Cameras Camera [n] Focus Mode: Mode
where
  - Mode: Auto/Manual
    - Auto: The camera will auto focus once a call is connected, as well as after moving the camera (pan, tilt, zoom). The system will use auto focus only for a few seconds to set the right focus; then auto focus is turned off to prevent continuous focus adjustments of the camera.
    - Manual: Turn the autofocus off and adjust the camera focus manually.
xConfiguration Cameras Camera [1] Mirror

With Mirror mode (horizontal flip) you can mirror the image on screen. Mirroring applies both to the self-view and the video that is transmitted to the far end.

Requires user role: USER
Default value: Auto

**USAGE:**

```bash
xConfiguration Cameras Camera [n] Mirror: Mirror
```

where

- **Mirror**: Auto/Off/On
  - Auto: If the camera detects that it is mounted upside down, the image is automatically mirrored. If the camera cannot auto-detect whether it is mounted upside down or not, the image is not changed.
  - Off: Display the image as other people see you.
  - On: Display the image as you see yourself in a mirror.

xConfiguration Cameras Camera [1] Whitebalance Mode

Define the camera white balance mode.

Requires user role: USER
Default value: Auto

**USAGE:**

```bash
xConfiguration Cameras Camera [n] Whitebalance Mode: Mode
```

where

- **Mode**: Auto/Manual
  - Auto: The camera will continuously adjust the white balance depending on the camera view.

xConfiguration Cameras Camera [1] Whitebalance Level

Define the white balance level. Requires the Cameras Camera [n] Whitebalance Mode to be set to manual.

Requires user role: USER
Default value: 1

**USAGE:**

```bash
xConfiguration Cameras Camera [n] Whitebalance Level: Level
```

where

- **Level**: Integer (1..16)
  - The white balance level.
Conference configuration

xConfiguration Conference AutoAnswer Mode

Define the auto answer mode. Use the Conference AutoAnswer Delay setting if you want the system to wait a number of seconds before answering the call, and use the Conference AutoAnswer Mute setting if you want your microphone to be muted when the call is answered.

Requires user role: ADMIN
Default value: Off

USAGE:
xConfiguration Conference AutoAnswer Mode: Mode
where
  Mode: Off/On
    Off: You must answer incoming calls manually by pressing the OK key or the green Call key on the remote control, or by tapping Answer on the Touch controller.
    On: The system automatically answers incoming calls, except if you are already in a call. You must always answer or decline incoming calls manually when you are already engaged in a call.

xConfiguration Conference AutoAnswer Mute

Define if the microphone shall be muted when an incoming call is automatically answered. Requires that AutoAnswer Mode is switched on.

Requires user role: ADMIN
Default value: Off

USAGE:
xConfiguration Conference AutoAnswer Mute: Mute
where
  Mute: Off/On
    Off: The incoming call will not be muted.
    On: The incoming call will be muted when automatically answered.

xConfiguration Conference AutoAnswer Delay

Define how long (in seconds) an incoming call has to wait before it is answered automatically by the system. Requires that AutoAnswer Mode is switched on.

Requires user role: ADMIN
Default value: 0

USAGE:
xConfiguration Conference AutoAnswer Delay: Delay
where
  Delay: Integer (0..50)
    The auto answer delay (seconds).

xConfiguration Conference DefaultCall Rate

Define the Default Call Rate to be used when placing calls from the system.

Requires user role: ADMIN
Default value: 1920

USAGE:
xConfiguration Conference DefaultCall Rate: Rate
where
  Rate: Integer (64..3072)
xConfiguration Conference DoNotDisturb DefaultTimeout

This setting determines the default duration of a Do Not Disturb session, i.e. the period when incoming calls are rejected and registered as missed calls. The session can be terminated earlier by using the user interface (remote control or Touch controller). The default value is 60 minutes.

Requires user role: ADMIN
Default value: 60

**USAGE:**

xConfiguration Conference DoNotDisturb DefaultTimeout: _DefaultTimeout_
where

_DEFAULTTimeout_: Integer (0..1440)

Set the number of minutes (between 0 and 1440, i.e. 24 hours) before the Do Not Disturb session times out automatically.

xConfiguration Conference Encryption Mode

Lets you decide if the remote side (far end) should be allowed to select your video sources and control your local camera (pan, tilt, zoom).

Requires user role: ADMIN
Default value: BestEffort

**USAGE:**

xConfiguration Conference Encryption Mode: _Mode_
where

_MODE_: Off/On/BestEffort

Off: The system will not use encryption.
On: The system will only allow calls that are encrypted.
BestEffort: The system will use encryption whenever possible.

> In Point to point calls: If the far end system supports encryption (AES-128), the call will be encrypted. If not, the call will proceed without encryption.

> In MultiSite calls: In order to have encrypted MultiSite conferences, all sites must support encryption. If not, the conference will be unencrypted.

xConfiguration Conference MaxReceiveCallRate

Define the maximum receive bit rate to be used when placing or receiving calls. Note that this is the maximum bit rate for each individual call; use the Conference MaxTotalReceiveCallRate setting to set the aggregated maximum for all simultaneous active calls.

Requires user role: ADMIN
Default value: 3072

**USAGE:**

xConfiguration Conference MaxReceiveCallRate: _MaxReceiveCallRate_
where

_MAXReceiveCallRate_: Integer (64..3072)

Set the maximum receive call rate (kbps).
xConfiguration Conference MaxTransmitCallRate

Define the maximum transmit bit rate to be used when placing or receiving calls. Note that this is the maximum bit rate for each individual call; use the Conference MaxTotalTransmitCallRate setting to set the aggregated maximum for all simultaneous active calls.

Requires user role: ADMIN
Default value: 3072

**USAGE:**
xConfiguration Conference MaxTransmitCallRate: MaxTransmitCallRate

where

MaxTransmitCallRate: Integer (64..3072)

Set the maximum transmit call rate (kbps).

xConfiguration Conference MaxTotalTransmitCallRate

Define the maximum overall transmit bit rate allowed. This product does not support multiple simultaneous calls, so the total transmit call rate will be the same as the transmit bit rate for one call (ref. Conference MaxTransmitCallRate setting).

Requires user role: ADMIN
Default value: 3072

**USAGE:**
xConfiguration Conference MaxTotalTransmitCallRate: MaxTotalTransmitCallRate

where

MaxTotalTransmitCallRate: Integer (64..3072)

Set the maximum transmit call rate (kbps).

xConfiguration Conference MaxTotalReceiveCallRate

Define the maximum overall receive bit rate allowed. This product does not support multiple simultaneous calls, so the total receive call rate will be the same as the receive bit rate for one call (ref. Conference MaxReceiveCallRate setting).

Requires user role: ADMIN
Default value: 3072

**USAGE:**
xConfiguration Conference MaxTotalReceiveCallRate: MaxTotalReceiveCallRate

where

MaxTotalReceiveCallRate: Integer (64..3072)

Set the maximum receive call rate (kbps).
Network configuration

xConfiguration Network [1] DNS Domain Name
The DNS Domain Name is the default domain name suffix which is added to unqualified names.
Example: If the DNS Domain Name is "company.com" and the name to lookup is "MyVideoSystem", this will result in the DNS lookup "MyVideoSystem.company.com".
Requires user role: ADMIN
Default value: ""

USAGE:
xConfiguration Network n DNS Domain Name: "Name"
where
n: Index that identifies the network. Range: 1..1
Name: String (0, 64)
The DNS domain name.

Define the network addresses for DNS servers. Up to three addresses may be specified. If the network addresses are unknown, contact your administrator or Internet Service Provider.
Requires user role: ADMIN
Default value: ""

USAGE:
xConfiguration Network n DNS Server m Address: "Address"
where
n: Index that identifies the network. Range: 1..1
m: Index that identifies the DNS server. Maximum three DNS servers are allowed. Range: 1..3
Address: String (0, 64)
A valid IPv4 address or IPv6 address.

xConfiguration Network [1] IEEE8021X Mode
The system can be connected to an IEEE 802.1X LAN network, with a port-based network access control that is used to provide authenticated network access for Ethernet networks.
Requires user role: ADMIN
Default value: Off

USAGE:
xConfiguration Network n IEEE8021X Mode: Mode
where
n: Index that identifies the network. Range: 1..1
Mode: Off/On
   Off: The 802.1X authentication is disabled (default).
   On: The 802.1X authentication is enabled.

xConfiguration Network [1] IEEE8021X TlsVerify
Verification of the server-side certificate of an IEEE802.1x connection against the certificates in the local CA-list when TLS is used. The CA-list must be uploaded to the video system. This can be done from the web interface.
This setting takes effect only when Network [1] IEEE8021X Eap Tls is enabled (On).
Requires user role: ADMIN
Default value: Off

USAGE:
xConfiguration Network n IEEE8021X TlsVerify: TlsVerify
where
n: Index that identifies the network. Range: 1..1
TlsVerify: Off/On
   Off: When set to Off, TLS connections are allowed without verifying the server-side X.509 certificate against the local CA-list. This should typically be selected if no CA-list has been uploaded to the codec.
   On: When set to On, the server-side X.509 certificate will be validated against the local CA-list for all TLS connections. Only servers with a valid certificate will be allowed.
API Reference Guide

xConfiguration Network [1] IEEE8021X UseClientCertificate

Authentication using a private key/certificate pair during an IEEE802.1x connection. The authentication X.509 certificate must be uploaded to the video system. This can be done from the web interface.

Requires user role: ADMIN
Default value: Off

** USAGE:**

```

where

  n: Index that identifies the network. Range: 1..1

UseClientCertificate: Off/On

  Off: When set to Off client-side authentication is not used (only server-side).
  On: When set to On the client (video system) will perform a mutual authentication TLS handshake with the server.
```

xConfiguration Network [1] IEEE8021X Identity

Define the user name for 802.1X authentication.

Requires user role: ADMIN
Default value: ""

** USAGE:**

```

where

  n: Index that identifies the network. Range: 1..1

Identity: String (0, 64)

  The user name for 802.1X authentication.
```

xConfiguration Network [1] IEEE8021X Password

Define the password for 802.1X authentication.

Requires user role: ADMIN
Default value: ""

** USAGE:**

```

where

  n: Index that identifies the network. Range: 1..1
  Password: String (0, 32)

  The password for 802.1X authentication.
```


The 802.1X Anonymous ID string is to be used as unencrypted identity with EAP (Extensible Authentication Protocol) types that support different tunnelled identity, like EAP-PEAP and EAP-TTLS. If set, the anonymous ID will be used for the initial (uncrypted) EAP Identity Request.

Requires user role: ADMIN
Default value: ""

** USAGE:**

```

where

  n: Index that identifies the network. Range: 1..1
  AnonymousIdentity: String (0, 64)

  The 802.1X Anonymous ID string.
```
Define the Md5 (Message-Digest Algorithm 5) mode. This is a Challenge Handshake Authentication Protocol that relies on a shared secret. Md5 is a Weak security.

Requires user role: ADMIN
Default value: On

**USAGE:**
```
xConfiguration Network [1] IEEE8021X Eap Md5: *Md5*
```
where
- `Md5`: Index that identifies the network. Range: 1..1
  - Off: The EAP-MD5 protocol is disabled.
  - On: The EAP-MD5 protocol is enabled (default).

xConfiguration Network [1] IEEE8021X Eap Ttls
Define the TTLS (Tunneled Transport Layer Security) mode. Authenticates LAN clients without the need for client certificates. Developed by Funk Software and Certicom. Usually supported by Agere Systems, Proxim and Avaya.

Requires user role: ADMIN
Default value: On

**USAGE:**
```
xConfiguration Network [1] IEEE8021X Eap Ttls: *Ttls*
```
where
- `Ttls`: Index that identifies the network. Range: 1..1
  - Off: The EAP-TTLS protocol is disabled.
  - On: The EAP-TTLS protocol is enabled (default).

xConfiguration Network [1] IEEE8021X Eap Tls
Enable or disable the use of EAP-TLS (Transport Layer Security) for IEEE802.1x connections. The EAP-TLS protocol, defined in RFC 5216, is considered one of the most secure EAP standards. LAN clients are authenticated using client certificates.

Requires user role: ADMIN
Default value: On

**USAGE:**
```
xConfiguration Network [1] IEEE8021X Eap Tls: *Tls*
```
where
- `Tls`: Index that identifies the network. Range: 1..1
  - Off: The EAP-TLS protocol is disabled.
  - On: The EAP-TLS protocol is enabled (default).

xConfiguration Network [1] IEEE8021X Eap Peap
Define the Peap (Protected Extensible Authentication Protocol) mode. Authenticates LAN clients without the need for client certificates. Developed by Microsoft, Cisco and RSA Security.

Requires user role: ADMIN
Default value: On

**USAGE:**
```
xConfiguration Network [1] IEEE8021X Eap Peap: *Peap*
```
where
- `Peap`: Index that identifies the network. Range: 1..1
  - Off: The EAP-PEAP protocol is disabled.
  - On: The EAP-PEAP protocol is enabled (default).
xConfiguration Network [1] IPStack
Select if the system should use IPv4, IPv6, or dual IP stack, on the network interface. NOTE: After changing this setting you may have to wait up to 30 seconds before it takes effect.

Requires user role: ADMIN
Default value: Dual

**USAGE:**
xConfiguration Network [1] IPStack: IPStack
where
- n: Index that identifies the network. Range: 1..1
- IPStack: Dual/IPv4/IPv6
  - Dual: When set to Dual, the network interface can operate on both IP versions at the same time, and can have both an IPv4 and an IPv6 address at the same time.
  - IPv4: When set to IPv4, the system will use IPv4 on the network interface.
  - IPv6: When set to IPv6, the system will use IPv6 on the network interface.

xConfiguration Network [1] IPv4 Assignment
Define how the system will obtain its IPv4 address, subnet mask and gateway address. This setting only applies to systems on IPv4 networks.

Requires user role: ADMIN
Default value: DHCP

**USAGE:**
xConfiguration Network [1] IPv4 Assignment: Assignment
where
- n: Index that identifies the network. Range: 1..1
- Assignment: Static/DHCP
  - Static: The addresses must be configured manually using the Network IPv4 Address, Network IPv4 Gateway and Network IPv4 SubnetMask settings (static addresses).
  - DHCP: The system addresses are automatically assigned by the DHCP server.

xConfiguration Network [1] IPv4 Address
Define the static IPv4 network address for the system. Applicable only when Network IPv4 Assignment is set to Static.

Requires user role: ADMIN
Default value: ""

**USAGE:**
xConfiguration Network [1] IPv4 Address: "Address"
where
- n: Index that identifies the network. Range: 1..1
- Address: String (0, 64)
  - A valid IPv4 address.

xConfiguration Network [1] IPv4 Gateway
Define the IPv4 network gateway address. Applicable only when the Network IPv4 Assignment is set to Static.

Requires user role: ADMIN
Default value: ""

**USAGE:**
xConfiguration Network [1] IPv4 Gateway: "Gateway"
where
- n: Index that identifies the network. Range: 1..1
- Gateway: String (0, 64)
  - A valid IPv4 address.
**xConfiguration Network [1] IPv4 SubnetMask**

Define the IPv4 network subnet mask. Applicable only when the Network IPv4 Assignment is set to Static.

Requires user role: ADMIN

Default value: **"**

**USAGE:**

```
xConfiguration Network n IPv4 SubnetMask: "SubnetMask"
```

where

- **n**: Index that identifies the network. Range: 1..1
- **SubnetMask**: String (0, 64)

A valid IPv4 address.

**xConfiguration Network [1] IPv6 Assignment**

Define how the system will obtain its IPv6 address and the default gateway address. This setting only applies to systems on IPv6 networks.

Requires user role: ADMIN

Default value: Autoconf

**USAGE:**

```
xConfiguration Network n IPv6 Assignment: Assignment
```

where

- **n**: Index that identifies the network. Range: 1..1
- **Assignment**: Static/DHCPv6/Autoconf

**Static**: The codec and gateway IP addresses must be configured manually using the Network IPv6 Address and Network IPv6 Gateway settings. The options, for example NTP and DNS server addresses, must either be set manually or obtained from a DHCPv6 server. The Network IPv6 DHCPOptions setting determines which method to use.

DHCPv6: All IPv6 addresses, including options, will be obtained from a DHCPv6 server. See RFC 3315 for a detailed description. The Network IPv6 DHCPOptions setting will be ignored.

Autoconf: Enable IPv6 stateless autoconfiguration of the IPv6 network interface. See RFC 4862 for a detailed description. The options, for example NTP and DNS server addresses, must either be set manually or obtained from a DHCPv6 server. The Network IPv6 DHCPOptions setting determines which method to use.

**xConfiguration Network [1] IPv6 Address**

Define the static IPv6 network address for the system. Applicable only when the Network IPv6 Assignment is set to Static.

Requires user role: ADMIN

Default value: **"**

**USAGE:**

```
xConfiguration Network n IPv6 Address: "Address"
```

where

- **n**: Index that identifies the network. Range: 1..1
- **Address**: String (0, 64)

A valid IPv6 address.

**xConfiguration Network [1] IPv6 Gateway**

Define the IPv6 network gateway address. This setting is only applicable when the Network IPv6 Assignment is set to Static.

Requires user role: ADMIN

Default value: **"**

**USAGE:**

```
xConfiguration Network n IPv6 Gateway: "Gateway"
```

where

- **n**: Index that identifies the network. Range: 1..1
- **Gateway**: String (0, 64)

A valid IPv6 address.
xConfiguration Network [1] IPv6 DHCPOptions

Retrieve a set of DHCP options, for example NTP and DNS server addresses, from a DHCPv6 server.

Requires user role: ADMIN
Default value: On

**USAGE:**

```
xConfiguration Network [1] IPv6 DHCPOptions: DHCPOptions
```

where

- `n`: Index that identifies the network. Range: 1..1
- `DHCPOptions`: Off/On
  - Off: Disable the retrieval of DHCP options from a DHCPv6 server.
  - On: Enable the retrieval of a selected set of DHCP options from a DHCPv6 server.

xConfiguration Network [1] MTU

Define the Ethernet MTU (Maximum Transmission Unit).

Requires user role: ADMIN
Default value: 1500

**USAGE:**

```
xConfiguration Network [1] MTU: MTU
```

where

- `n`: Index that identifies the network. Range: 1..1
- `MTU`: Integer (576..1500)
  - Set a value for the MTU (bytes).

xConfiguration Network [1] QoS Mode

The QoS (Quality of Service) is a method which handles the priority of audio, video and data in the network. The QoS settings must be supported by the infrastructure. Diffserv (Differentiated Services) is a computer networking architecture that specifies a simple, scalable and coarse-grained mechanism for classifying, managing network traffic and providing QoS priorities on modern IP networks.

Requires user role: ADMIN
Default value: Diffserv

**USAGE:**

```
xConfiguration Network [1] QoS Mode: Mode
```

where

- `Mode`: Off/Diffserv
  - Off: No QoS method is used.
  - Diffserv: When you set the QoS Mode to Diffserv, the Network QoS Diffserv Audio, Network QoS Diffserv Video, Network QoS Diffserv Data, Network QoS Diffserv Signalling, Network QoS Diffserv ICMPv6 and Network QoS Diffserv NTP settings are used to prioritize packets.

xConfiguration Network [1] QoS Diffserv Audio

This setting will only take effect if Network QoS Mode is set to Diffserv.

Define which priority Audio packets should have in the IP network.

The priority for the packets ranges from 0 to 63 - the higher the number, the higher the priority. The recommended class for Audio is CS4, which equals the decimal value 32. If in doubt, contact your network administrator.

The priority set here might be overridden when packets are leaving the network controlled by the local network administrator.

Requires user role: ADMIN
Default value: 0

**USAGE:**

```
```

where

- `Audio`: Integer (0..63)
  - Set the priority of the audio packets in the IP network - the higher the number, the higher the priority. The default value is 0 (best effort).
xConfiguration Network [1] QoS Diffserv Video

This setting will only take effect if Network QoS Mode is set to Diffserv.

Define which priority Video packets should have in the IP network. The packets on the presentation channel (shared content) are also in the Video packet category. The priority for the packets ranges from 0 to 63 - the higher the number, the higher the priority. The recommended class for Video is CS4, which equals the decimal value 32. If in doubt, contact your network administrator.

The priority set here might be overridden when packets are leaving the network controlled by the local network administrator.

Requires user role: ADMIN
Default value: 0

USAGE:
xConfiguration Network [1] QoS Diffserv Video: Video
where
   Video: Integer (0..63)
   Set the priority of the video packets in the IP network - the higher the number, the higher the priority. The default value is 0 (best effort).

xConfiguration Network [1] QoS Diffserv Data

This setting will only take effect if Network QoS Mode is set to Diffserv.

Define which priority Data packets should have in the IP network.

The priority for the packets ranges from 0 to 63 - the higher the number, the higher the priority. The recommended value for Data is 0, which means best effort. If in doubt, contact your network administrator.

The priority set here might be overridden when packets are leaving the network controlled by the local network administrator.

Requires user role: ADMIN
Default value: 0

USAGE:
xConfiguration Network [1] QoS Diffserv Data: Data
where
   Data: Integer (0..63)
   Set the priority of the data packets in the IP network - the higher the number, the higher the priority. The default value is 0 (best effort).

xConfiguration Network [1] QoS Diffserv Signalling

This setting will only take effect if Network QoS Mode is set to Diffserv.

Define which priority Signalling packets that are deemed critical (time-sensitive) for the real-time operation should have in the IP network.

The priority for the packets ranges from 0 to 63 - the higher the number, the higher the priority. The recommended class for Signalling is CS3, which equals the decimal value 24. If in doubt, contact your network administrator.

The priority set here might be overridden when packets are leaving the network controlled by the local network administrator.

Requires user role: ADMIN
Default value: 0

USAGE:
xConfiguration Network [1] QoS Diffserv Signalling: Signalling
where
   Signalling: Integer (0..63)
   Set the priority of the signalling packets in the IP network - the higher the number, the higher the priority. The default value is 0 (best effort).

xConfiguration Network [1] QoS Diffserv ICMPv6

This setting will only take effect if Network QoS Mode is set to Diffserv.

Define which priority ICMPv6 packets should have in the IP network.

The priority for the packets ranges from 0 to 63 - the higher the number, the higher the priority. The recommended value for ICMPv6 is 0, which means best effort. If in doubt, contact your network administrator.

The priority set here might be overridden when packets are leaving the network controlled by the local network administrator.

Requires user role: ADMIN
Default value: 0

USAGE:
where
   ICMPv6: Integer (0..63)
   Set the priority of the ICMPv6 packets in the IP network - the higher the number, the higher the priority. The default value is 0 (best effort).
**xConfiguration Network [1] QoS Diffserv NTP**

This setting will only take effect if Network QoS Mode is set to Diffserv.

Define which priority NTP packets should have in the IP network.

The priority for the packets ranges from 0 to 63 - the higher the number, the higher the priority. The recommended value for NTP is 0, which means best effort. If in doubt, contact your network administrator.

The priority set here might be overridden when packets are leaving the network controlled by the local network administrator.

Requires user role: ADMIN

Default value: 0

**USAGE:**

```
```

where

- **NTP**: Integer (0..63)

  Set the priority of the NTP packets in the IP network - the higher the number, the higher the priority. The default value is 0 (best effort).

**xConfiguration Network [1] RemoteAccess Allow**

Define which IP addresses (IPv4/IPv6) are allowed for remote access to the codec from SSH/Telnet/HTTP/HTTPS. Multiple IP addresses are separated by a white space.

A network mask (IP range) is specified by <ip address>/N, where N is 1-32 for IPv4, and N is 1-128 for IPv6. The /N is a common indication of a network mask where the first N bits are set. Thus 192.168.0.0/24 would match any address starting with 192.168.0, since these are the first 24 bits in the address.

Requires user role: ADMIN

Default value: ""

**USAGE:**

```
```

where

- **n**: Index that identifies the network. Range: 1..1

  - **Allow**: String (0, 255)

    A valid IPv4 address or IPv6 address.

**xConfiguration Network [1] VLAN Voice Mode**

Define the VLAN voice mode. The VLAN Voice Mode will be set to Auto automatically if you have Cisco UCM (Cisco Unified Communications Manager) as provisioning infrastructure. Note that Auto mode will NOT work if the NetworkServices CDP Mode setting is Off.

Requires user role: ADMIN

Default value: Auto

**USAGE:**

```
xConfiguration Network [1] VLAN Voice Mode: Mode
```

where

- **Mode**: Auto/Manual/Off

  - **Auto**: The Cisco Discovery Protocol (CDP), if available, assigns an id to the voice VLAN. If CDP is not available, VLAN is not enabled.

  - **Manual**: The VLAN ID is set manually using the Network VLAN Voice VlanId setting. If CDP is available, the manually set value will be overruled by the value assigned by CDP.

  - **Off**: VLAN is not enabled.

**xConfiguration Network [1] VLAN Voice VlanId**

Define the VLAN voice ID. This setting will only take effect if Network VLAN Voice Mode is set to Manual.

Requires user role: ADMIN

Default value: 1

**USAGE:**

```
xConfiguration Network [1] VLAN Voice VlanId: VlanId
```

where

- **n**: Index that identifies the network. Range: 1..1

  - **VlanId**: Integer (1..4094)

    Set the VLAN voice ID.
NetworkServices configuration

xConfiguration NetworkServices CDP Mode
Enable or disable the CDP (Cisco Discovery Protocol) daemon. Enabling CDP will make the
endpoint report certain statistics and device identifiers to a CDP-enabled switch. If CDP is
disabled, the Network VLAN Voice Mode: Auto setting will not work.

Requires user role: ADMIN
Default value: On

USAGE:
```
xConfiguration NetworkServices CDP Mode: Mode
```
where

**Mode**: Off/On
- Off: The CDP daemon is disabled.
- On: The CDP daemon is enabled.

xConfiguration NetworkServices HTTP Mode
Define whether or not to allow access to the video system using the HTTP or HTTPS (HTTP
Secure) protocols. Note that the video system’s web interface use HTTP or HTTPS. If this
setting is switched Off, you cannot use the web interface.

If you need extra security (encryption and decryption of requests, and pages that are returned
by the web server), allow only HTTPS.

Requires user role: ADMIN
Default value: HTTP+HTTPS

USAGE:
```
xConfiguration NetworkServices HTTP Mode: Mode
```
where

**Mode**: Off/HTTP+HTTPS/HTTPS
- Off: Access to the video system not allowed via HTTP or HTTPS.
- HTTP+HTTPS: Access to the video system allowed via both HTTP and HTTPS.
- HTTPS: Access to the video system allowed via HTTPS, but not via HTTP.

xConfiguration NetworkServices NTP Mode
The Network Time Protocol (NTP) is used to synchronize the system’s time and date to a refer-
ence time server. The time server will be queried regularly for time updates.

Requires user role: ADMIN
Default value: Auto

USAGE:
```
xConfiguration NetworkServices NTP Mode: Mode
```
where

**Mode**: Auto/Manual/Off
- Auto: The system will use an NTP server for time reference. As default, the server
  address will be obtained from the network's DHCP server. If a DHCP server is not used,
  or if the DHCP server does not provide an NTP server address, the NTP server address
  that is specified in the NetworkServices NTP Server [n] Address setting will be used.
- Manual: The system will use the NTP server that is specified in the NetworkServices
  NTP Server [n] Address setting for time reference.
- Off: The system will not use an NTP server. The NetworkServices NTP Server [n]
  Address setting will be ignored.

xConfiguration NetworkServices NTP Server [1..3] Address
The address of the NTP server that will be used when NetworkServices NTP Mode is set to
Manual, and when NetworkServices NTP Mode is set to Auto and no address is supplied by a
DHCP server.

Requires user role: ADMIN
Default value: 0.tandberg.pool.ntp.org

USAGE:
```
xConfiguration NetworkServices NTP Server n Address: "Address"
```
where

**Address**: String (0, 64)
- A valid IPv4 address, IPv6 address or DNS name.
xConfiguration NetworkServices SIP Mode

Define whether the system should be able to place and receive SIP calls or not.

Requires user role: ADMIN
Default value: On

**USAGE:**

xConfiguration NetworkServices SIP Mode: Mode

where

Mode: Off/On

- Off: Disable the possibility to place and receive SIP calls.
- On: Enable the possibility to place and receive SIP calls (default).

xConfiguration NetworkServices SNMP Mode

SNMP (Simple Network Management Protocol) is used in network management systems to monitor network-attached devices (routers, servers, switches, projectors, etc) for conditions that warrant administrative attention. SNMP exposes management data in the form of variables on the managed systems, which describe the system configuration. These variables can then be queried (set to ReadOnly) and sometimes set (set to ReadWrite) by managing applications.

Requires user role: ADMIN
Default value: ReadOnly

**USAGE:**

xConfiguration NetworkServices SNMP Mode: Mode

where

Mode: Off/ReadOnly/ReadWrite

- Off: Disable the SNMP network service.
- ReadOnly: Enable the SNMP network service for queries only.
- ReadWrite: Enable the SNMP network service for both queries and commands.

xConfiguration NetworkServices SNMP Host [1..3] Address

Define the address of up to three SNMP Managers.

The system’s SNMP Agent (in the codec) responds to requests from SNMP Managers (a PC program etc.), for example about system location and system contact. SNMP traps are not supported.

Requires user role: ADMIN
Default value: ""

**USAGE:**

xConfiguration NetworkServices SNMP Host n Address: "Address"

where

n: Identifies the SNMP host. Maximum three SNMP hosts are allowed. Range: 1..3
Address: String (0, 64)

A valid IPv4 address, IPv6 address or DNS name.

xConfiguration NetworkServices SNMP CommunityName

Define the name of the Network Services SNMP Community. SNMP Community names are used to authenticate SNMP requests. SNMP requests must have a password (case sensitive) in order to receive a response from the SNMP Agent in the codec. The default password is “public”. If you have the Cisco TelePresence Management Suite (TMS) you must make sure the same SNMP Community is configured there too. NOTE: The SNMP Community password is case sensitive.

Requires user role: ADMIN
Default value: ""

**USAGE:**

xConfiguration NetworkServices SNMP CommunityName: "CommunityName"

where

CommunityName: String (0, 50)

The SNMP community name.
**xConfiguration NetworkServices SNMP SystemContact**

Define the name of the Network Services SNMP System Contact.

Requires user role: ADMIN
Default value: ""

**USAGE:**

```java
xConfiguration NetworkServices SNMP SystemContact: SystemContact
```

where

- **SystemContact**: String (0, 50)
- The name of the SNMP system contact.

**xConfiguration NetworkServices SNMP SystemLocation**

Define the name of the Network Services SNMP System Location.

Requires user role: ADMIN
Default value: ""

**USAGE:**

```java
xConfiguration NetworkServices SNMP SystemLocation: SystemLocation
```

where

- **SystemLocation**: String (0, 50)
- The name of the SNMP system location.

**xConfiguration NetworkServices SSH Mode**

SSH (or Secure Shell) protocol can provide secure encrypted communication between the codec and your local computer.

Requires user role: ADMIN
Default value: On

**USAGE:**

```java
xConfiguration NetworkServices SSH Mode: Mode
```

where

- **Mode**: Off/On
  - Off: The SSH protocol is disabled.
  - On: The SSH protocol is enabled.

**xConfiguration NetworkServices Telnet Mode**

Telnet is a network protocol used on the Internet or Local Area Network (LAN) connections.

Requires user role: ADMIN
Default value: Off

**USAGE:**

```java
xConfiguration NetworkServices Telnet Mode: Mode
```

where

- **Mode**: Off/On
  - Off: The Telnet protocol is disabled. This is the factory setting.
  - On: The Telnet protocol is enabled.

**xConfiguration NetworkServices WelcomeText**

Choose which information the user should see when logging on to the codec through Telnet/SSH.

Requires user role: ADMIN
Default value: On

**USAGE:**

```java
xConfiguration NetworkServices WelcomeText: WelcomeText
```

where

- **WelcomeText**: Off/On
  - Off: The welcome text is: Login successful
  - On: The welcome text is: Welcome to <system name>; Software version; Software release date; Login successful.
Peripherals configuration

xConfiguration Peripherals Profile TouchPanels
Define the number of touch panels that are expected to be connected to the video system. This information is used by the video system’s diagnostics service. If the number of connected touch panels does not match this setting, the diagnostics service will report it as an inconsistency. Note that only one Cisco Touch controller is supported in this version.

Requires user role: ADMIN
Default value: NotSet

**USAGE:**
xConfiguration Peripherals Profile TouchPanels: TouchPanels
where

TouchPanels: NotSet/Minimum1/0/1/2/3/4/5
NotSet: No touch panel check is performed.
Minimum1: At least one touch panel should be connected to the video system.
0-5: Select the number of Touch controllers that are expected to be connected to the video system.

Phonebook configuration

xConfiguration Phonebook Server [1] ID
Define a name for the external phone book.
Requires user role: ADMIN
Default value: ""

**USAGE:**
xConfiguration Phonebook Server [1] ID: "ID"
where

ID: String (0, 64)
The name for the external phone book.

xConfiguration Phonebook Server [1] Type
Select the phonebook server type.
Requires user role: ADMIN
Default value: Off

**USAGE:**
xConfiguration Phonebook Server [1] Type: Type
where

Type: Off/VCS/TMS/CUCM
Off: Do not use a phonebook.
VCS: The phonebook is located on the Cisco TelePresence Video Communication Server.
TMS: The phonebook is located on the Cisco TelePresence Management Suite server.
CUCM: The phonebook is located on the Cisco Unified Communications Manager.
xConfiguration Phonebook Server [1] URL
Define the address (URL) to the external phone book server.

Requires user role: ADMIN
Default value: ""

**USAGE:**
xConfiguration Phonebook Server n URL: "URL"
where

URL: String (0, 255)
A valid address (URL) to the phone book server.

Provisioning configuration

xConfiguration Provisioning Connectivity
This setting controls how the device discovers whether it should request an internal or external configuration from the provisioning server.

Requires user role: ADMIN
Default value: Auto

**USAGE:**
xConfiguration Provisioning Connectivity: Connectivity
where

Connectivity: Internal/External/Auto
  - Internal: Request internal configuration.
  - External: Request external configuration.
  - Auto: Automatically discover using NAPTR queries whether internal or external configurations should be requested. If the NAPTR responses have the "e" flag, external configurations will be requested. Otherwise internal configurations will be requested.
xConfiguration Provisioning Mode

It is possible to configure a video system using a provisioning system (external manager). This allows video conferencing network administrators to manage many video systems simultaneously. With this setting you choose which type of provisioning system to use. Provisioning can also be switched off. Contact your provisioning system provider/representative for more information.

Requires user role: ADMIN
Default value: Auto

**USAGE:**

```
xConfiguration Provisioning Mode: Mode
where
   Mode: Off/Auto/TMS/VCS/CUCM/Edge
```

- Off: The video system is not configured by a provisioning system.
- Auto: Automatically select the provisioning server.
- TMS: Push configurations to the video system from TMS (Cisco TelePresence Management System).
- VCS: Push configurations to the video system from VCS (Cisco TelePresence Video Communication Server).
- CUCM: Push configurations to the video system from CUCM (Cisco Unified Communications Manager).
- Edge: Push configurations to the video system from CUCM (Cisco Unified Communications Manager). The system connects to CUCM via the Collaboration Edge infrastructure.

xConfiguration Provisioning LoginName

This is the username part of the credentials used to authenticate the video system with the provisioning server. This setting must be used when required by the provisioning server.

Requires user role: ADMIN
Default value: ""

**USAGE:**

```
xConfiguration Provisioning LoginName: "LoginName"
```

where

- **LoginName**: String (0, 80)
- A valid username.

xConfiguration Provisioning Password

This is the password part of the credentials used to authenticate the video system with the provisioning server. This setting must be used when required by the provisioning server.

Requires user role: ADMIN
Default value: ""

**USAGE:**

```
xConfiguration Provisioning Password: "Password"
```

where

- **Password**: String (0, 64)
- A valid password.
xConfiguration Provisioning HttpMethod

Select the HTTP method to be used for the provisioning.

Requires user role: ADMIN
Default value: POST

USAGE:
xConfiguration Provisioning HttpMethod: HttpMethod
where
   HttpMethod: GET/POST
      GET: Select GET when the provisioning server supports GET.
      POST: Select POST when the provisioning server supports POST.

xConfiguration Provisioning ExternalManager Address

Define the IP Address or DNS name of the external manager / provisioning system.
If an External Manager Address (and Path) is configured, the system will send a message to this address when starting up. When receiving this message the external manager / provisioning system can return configurations/commands to the unit as a result.
When using CUCM or TMS provisioning, the DHCP server can be set up to provide the external manager address automatically (DHCP Option 242 for TMS, and DHCP Option 150 for CUCM). An address set in the Provisioning ExternalManager Address setting will override the address provided by DHCP.

Requires user role: ADMIN
Default value: ""

USAGE:
xConfiguration Provisioning ExternalManager Address: "Address"
where
   Address: String (0, 64)
      A valid IPv4 address, IPv6 address or DNS name.

xConfiguration Provisioning ExternalManager AlternateAddress

Only applicable when the endpoint is provisioned by Cisco Unified Communication Manager (CUCM) and an alternate CUCM is available for redundancy. Define the address of the alternate CUCM. If the main CUCM is not available, the endpoint will be provisioned by the alternate CUCM. When the main CUCM is available again, the endpoint will be provisioned by this CUCM.

Requires user role: ADMIN
Default value: ""

USAGE:
xConfiguration Provisioning ExternalManager AlternateAddress: "AlternateAddress"
where
   AlternateAddress: String (0, 64)
      A valid IPv4 address, IPv6 address or DNS name.

xConfiguration Provisioning ExternalManager Protocol

Define whether to use the HTTP (unsecure communication) or HTTPS (secure communication) protocol when sending requests to the external manager / provisioning system. The chosen protocol must be enabled in the NetworkServices HTTP Mode setting.

Requires user role: ADMIN
Default value: HTTP

USAGE:
xConfiguration Provisioning ExternalManager Protocol: Protocol
where
   Protocol: HTTPS/HTTP
      HTTPS: Send requests via HTTPS.
      HTTP: Send requests via HTTP.


**Proximity configuration**

**xConfiguration Provisioning ExternalManager Path**

Define the Path to the external manager / provisioning system. This setting is required when several management services reside on the same server, i.e. share the same External Manager address.

Requires user role: ADMIN
Default value: ""

**USAGE:**

```
xConfiguration Provisioning ExternalManager Path: "Path"
where
Path: String (0, 255)
A valid path to the external manager or provisioning system.
```

**xConfiguration Provisioning ExternalManager Domain**

Define the SIP domain for the VCS provisioning server.

Requires user role: ADMIN
Default value: ""

**USAGE:**

```
xConfiguration Provisioning ExternalManager Domain: "Domain"
where
Domain: String (0, 64)
A valid domain name.
```

**xConfiguration Proximity Mode**

Determine whether the video system will emit ultrasound pairing messages or not. When the video system emits ultrasound, Proximity clients can detect that they are close to the video system. In order to use a client, at least one of the Proximity services must be enabled (refer to the Proximity Services settings). In general, Cisco recommends enabling all the Proximity services.

Requires user role: ADMIN
Default value: Off

**USAGE:**

```
xConfiguration Proximity Mode: Mode
where
Mode: Off/On
Off: The video system does not emit ultrasound, and Proximity services cannot be used.
On: The video system emits ultrasound, and Proximity clients can detect that they are close to the video system. Enabled Proximity services can be used.
```

**xConfiguration Proximity Services CallControl**

Enable or disable basic call control features on Proximity clients. When this setting is enabled, you are able to control a call using a Proximity client (for example dial, mute, adjust volume and hang up). This service is supported by mobile devices (iOS and Android). Proximity Mode must be On for this setting to take any effect.

Requires user role: ADMIN
Default value: Disabled

**USAGE:**

```
xConfiguration Proximity Services CallControl: CallControl
where
CallControl: Enabled/Disabled
Enabled: Call control from a Proximity client is enabled.
Disabled: Call control from a Proximity client is disabled.
```
xConfiguration Proximity Services ContentShare FromClients
Enable or disable content sharing from Proximity clients. When this setting is enabled, you can share content from a Proximity client wirelessly on the video system, e.g. share your laptop screen. This service is supported by laptops (OS X and Windows). Proximity Mode must be On for this setting to take any effect.

Requires user role: ADMIN
Default value: Disabled

USAGE:
xConfiguration Proximity Services ContentShare FromClients: FromClients
where
  FromClients: Enabled/Disabled
          Enabled: Content sharing from a Proximity client is enabled.
          Disabled: Content sharing from a Proximity client is disabled.

xConfiguration Proximity Services ContentShare ToClients
Enable or disable content sharing to Proximity clients. When enabled, Proximity clients will receive the presentation from the video system. You can zoom in on details, view previous content and take snapshots. This service is supported by mobile devices (iOS and Android). Proximity Mode must be On for this setting to take any effect.

Requires user role: ADMIN
Default value: Disabled

USAGE:
xConfiguration Proximity Services ContentShare ToClients: ToClients
where
  ToClients: Enabled/Disabled
          Enabled: Content sharing to a Proximity client is enabled.
          Disabled: Content sharing to a Proximity client is disabled.

SerialPort configuration
xConfiguration SerialPort Mode
Enable/disable the serial port (connection via Micro USB to USB cable). The serial port uses 115200 bps, 8 data bits, no parity and 1 stop bit.

Requires user role: ADMIN
Default value: On

USAGE:
xConfiguration SerialPort Mode: Mode
where
  Mode: Off/On
          Off: Disable the serial port.
          On: Enable the serial port.

xConfiguration SerialPort LoginRequired
Define if login shall be required when connecting to the serial port.

Requires user role: ADMIN
Default value: On

USAGE:
xConfiguration SerialPort LoginRequired: LoginRequired
where
  LoginRequired: Off/On
          Off: The user can access the codec via the serial port without any login.
          On: Login is required when connecting to the codec via the serial port.
SIP configuration

xConfiguration SIP Authentication UserName
This is the user name part of the credentials used to authenticate towards the SIP proxy.

Requires user role: ADMIN
Default value: ""

**USAGE:**

```
xConfiguration SIP Authentication UserName: "UserName"
```

where

```
UserName: String (0, 128)
```
A valid username.

xConfiguration SIP Authentication Password
This is the password part of the credentials used to authenticate towards the SIP proxy.

Requires user role: ADMIN
Default value: ""

**USAGE:**

```
xConfiguration SIP Authentication Password: "Password"
```

where

```
Password: String (0, 128)
```
A valid password.

xConfiguration SIP DefaultTransport
Select the transport protocol to be used over the LAN.

Requires user role: ADMIN
Default value: Auto

**USAGE:**

```
xConfiguration SIP DefaultTransport: DefaultTransport
```

where

```
DefaultTransport: TCP/UDP/Tls/Auto
```

- **TCP:** The system will always use TCP as the default transport method.
- **UDP:** The system will always use UDP as the default transport method.
- **Tls:** The system will always use TLS as the default transport method. For TLS connections a SIP CA-list can be uploaded to the video system. If no such CA-list is available on the system then anonymous Diffie Hellman will be used.
- **Auto:** The system will try to connect using transport protocols in the following order: TLS, TCP, UDP.

xConfiguration SIP DisplayName
When configured the incoming call will report the display name instead of the SIP URI.

Requires user role: ADMIN
Default value: ""

**USAGE:**

```
xConfiguration SIP DisplayName: DisplayName
```

where

```
DisplayName: String (0, 255)
```
The name to be displayed instead of the SIP URI.
xConfiguration SIP Ice Mode
ICE (Interactive Connectivity Establishment, RFC 5245) is a NAT traversal solution that the endpoints can use to discover the optimized media path. Thus the shortest route for audio and video is always secured between the endpoints. NOTE: ICE is not supported when registered to CUCM (Cisco Unified Communication Manager).

Requires user role: ADMIN
Default value: Auto

**USAGE:**
xConfiguration SIP Ice Mode:

```
Mode
```

where

- **Mode**: Auto/Off/On
  - Auto: When set to Auto, ICE will be enabled if a turn server is provided, otherwise ICE will be disabled.
  - Off: Set to Off to disable ICE.
  - On: Set to On to enable ICE.

xConfiguration SIP Ice DefaultCandidate
This is the default IP address that the endpoint will receive media on until ICE has reached a conclusion about which media route to use (up to the first 5 seconds of a call).

Requires user role: ADMIN
Default value: Host

**USAGE:**
xConfiguration SIP Ice DefaultCandidate:

```
DefaultCandidate
```

where

- **DefaultCandidate**: Host/Rflx/Relay
  - Host: The endpoint will receive media on its own IP address.
  - Rflx: The endpoint will receive media on its public IP address as seen by the TURN server.
  - Relay: The endpoint will receive media on the IP address and port allocated on the TURN server, and is used as a fallback until ICE has concluded.

xConfiguration SIP ListenPort
Turn on or off the listening for incoming connections on the SIP TCP/UDP ports. If turned off, the endpoint will only be reachable through the SIP registrar (CUCM or VCS). It is recommended to leave this setting at its default value.

Requires user role: ADMIN
Default value: On

**USAGE:**
xConfiguration SIP ListenPort:

```
ListenPort
```

where

- **ListenPort**: Off/On
  - Off: Listening for incoming connections on the SIP TCP/UDP ports is turned off.
  - On: Listening for incoming connections on the SIP TCP/UDP ports is turned on.

xConfiguration SIP Proxy [1..4] Address
The Proxy Address is the manually configured address for the outbound proxy. It is possible to use a fully qualified domain name, or an IP address. The default port is 5060 for TCP and UDP but another one can be provided.

Requires user role: ADMIN
Default value: ""

**USAGE:**
xConfiguration SIP Proxy _n_ Address:

```
Address
```

where

- **n**: Index that identifies the proxy (maximum 4 proxys can be defined). Range: 1..4
- **Address**: String (0, 255)
  - A valid IPv4 address, IPv6 address or DNS name.
xConfiguration SIP Turn Server
Define the address of the TURN (Traversal Using Relay NAT) server. It is used as a media relay fallback and it is also used to discover the endpoint’s own public IP address.

Requires user role: ADMIN
Default value: ""

USAGE:
xConfiguration SIP Turn Server: "Server"
where
   Server: String (0, 255)
       The preferred format is DNS SRV record (e.g. _turn._udp.<domain>), or it can be a valid IPv4 or IPv6 address.

xConfiguration SIP Turn UserName
Define the user name needed for accessing the TURN server.

Requires user role: ADMIN
Default value: ""

USAGE:
xConfiguration SIP Turn UserName: "UserName"
where
   UserName: String (0, 128)
       A valid user name.

xConfiguration SIP Turn Password
Define the password needed for accessing the TURN server.

Requires user role: ADMIN
Default value: ""

USAGE:
xConfiguration SIP Turn Password: "Password"
where
   Password: String (0, 128)
       A valid password.

xConfiguration SIP URI
The SIP URI (Uniform Resource Identifier) is the address that is used to identify the video system. The URI is registered and used by the SIP services to route inbound calls to the system. The SIP URI syntax is defined in RFC 3261.

Requires user role: ADMIN
Default value: ""

USAGE:
xConfiguration SIP URI: "URI"
where
   URI: String (0, 255)
       An address (URI) that is compliant with the SIP URI syntax.
Standby configuration

xConfiguration Standby Control
Define whether the system should go into standby mode or not.
Requires user role: ADMIN
Default value: On

**USAGE:**
xConfiguration Standby Control: *Control* where
  - Off: The system will not enter standby mode.
  - On: The system will enter standby mode when the Standby Delay has timed out.
  Requires the Standby Delay to be set to an appropriate value.

xConfiguration Standby Delay
Define how long (in minutes) the system shall be in idle mode before it goes into standby mode.
Requires the Standby Control to be enabled.
Requires user role: ADMIN
Default value: 10

**USAGE:**
xConfiguration Standby Delay: *Delay* where
  - Delay: Integer (1..480)
    Set the standby delay (minutes).

xConfiguration Standby BootAction
Define the camera position after a restart of the codec.
Requires user role: ADMIN
Default value: DefaultCameraPosition

**USAGE:**
xConfiguration Standby BootAction: *BootAction*
  - BootAction: None/RestoreCameraPosition/DefaultCameraPosition
    - None: No action.
    - RestoreCameraPosition: When the video system restarts, the camera returns to the position that it had before the restart.
    - DefaultCameraPosition: When the video system restarts, the camera moves to the factory default position.

xConfiguration Standby StandbyAction
Define the camera position when going into standby mode.
Requires user role: ADMIN
Default value: PrivacyPosition

**USAGE:**
xConfiguration Standby StandbyAction: *StandbyAction*
  - StandbyAction: None/PrivacyPosition
    - None: No action.
    - PrivacyPosition: When the video system enters standby, the camera turns to a sideways position for privacy.
**xConfiguration Standby WakeupAction**

Define the camera position when leaving standby mode.

Requires user role: ADMIN

Default value: RestoreCameraPosition

**USAGE:**

```plaintext
xConfiguration Standby WakeupAction: WakeupAction
  where
    WakeupAction: None/RestoreCameraPosition/DefaultCameraPosition
    None: No action.
    RestoreCameraPosition: When the video system leaves standby, the camera returns to
                           the position that it had before entering standby.
    DefaultCameraPosition: When the video system leaves standby, the camera moves to
                           the factory default position.
```

**SystemUnit configuration**

**xConfiguration SystemUnit Name**

Define the system name. The system name will be sent as the hostname in a DHCP request and when the codec is acting as an SNMP Agent.

Requires user role: ADMIN

Default value: ""

**USAGE:**

```plaintext
xConfiguration SystemUnit Name: "Name"
  where
    Name: String (0, 50)
    Define the system name.
```
Time configuration

xConfiguration Time TimeFormat
Define the time format.
Requires user role: USER
Default value: 24H

**USAGE:**
xConfiguration Time TimeFormat: \texttt{TimeFormat}

where
\texttt{TimeFormat}: 24H/12H

- 24H: Set the time format to 24 hours.
- 12H: Set the time format to 12 hours (AM/PM).

xConfiguration Time DateFormat
Define the date format.
Requires user role: USER
Default value: DD_MM_YY

**USAGE:**
xConfiguration Time DateFormat: \texttt{DateFormat}

where
\texttt{DateFormat}: DD_MM_YY/MM_DD_YY/YY_MM_DD

- DD_MM_YY: The date January 30th 2010 will be displayed: 30.01.10
- MM_DD_YY: The date January 30th 2010 will be displayed: 01.30.10
- YY_MM_DD: The date January 30th 2010 will be displayed: 10.01.30

xConfiguration Time Zone
Define the time zone for the geographical location of the video system. The information in the value space is from the tz database, also called the IANA Time Zone Database.

Requires user role: USER
Default value: Etc/UTC

**USAGE:**
xConfiguration Time Zone: \texttt{Zone}

where
Select a time zone from the list.
UserInterface configuration

xConfiguration UserInterface ContactInfo Type
Choose which type of contact information to show in the status field in the upper left corner of the display and Touch controller.

Requires user role: ADMIN
Default value: Auto

USAGE:
xConfiguration UserInterface ContactInfo Type: Type
where
   Type: Auto/None/IPv4/IPv6/SipUri/SystemName/DisplayName
   Auto: Show the address which another system can dial to reach this system. The address depends on the system registration.
   None: Do not show any contact information.
   IPv4: Show the system's IPv4 address.
   IPv6: Show the system's IPv6 address.
   SipUri: Show the system's SIP URI (refer to the SIP URI setting).
   SystemName: Show the system's name (refer to the SystemUnit Name setting).
   DisplayName: Show the system's display name (refer to the SIP DisplayName setting).

xConfiguration UserInterface KeyTones Mode
You can configure the system to make a keyboard click sound effect (key tone) when pressing a key on the remote control, or when typing text or numbers on the Touch controller.

Requires user role: USER
Default value: On

USAGE:
xConfiguration UserInterface KeyTones Mode: Mode
where
   Mode: Off/On
   Off: There is no key tone sound effect.
   On: The key tone sound effect is turned on.

xConfiguration UserInterface Language
Select the language to be used in menus and messages on the screen and Touch controller. The default language is English.

Requires user role: USER
Default value: English

USAGE:
xConfiguration UserInterface Language: Language
where
   Language: English/ChineseSimplified/ChineseTraditional/Catalan/Czech/Danish/Dutch/Finnish/French/German/Hungarian/Italian/Japanese/Korean/Norwegian/Polish/PortugueseBrazilian/Russian/Spanish/Swedish/Turkish/Arabic/Hebrew
   Select a language from the list.
xConfiguration UserInterface OSD Output

Define on which monitor the on-screen menus, information and indicators (OSD) should be displayed. The system supports only one monitor, so this value is fixed and cannot be changed.

Requires user role: ADMIN
Default value: 1

USAGE:
```
xConfiguration UserInterface OSD Output: Output
```
where
```
Output: 1
```

---

Video configuration

xConfiguration Video DefaultMainSource

Define which video input source shall be used as the main video source.

Requires user role: USER
Default value: 1

USAGE:
```
xConfiguration Video DefaultMainSource: DefaultMainSource
```
where
```
DefaultMainSource: 1
```
Set the source to be used as the main video source.

---

xConfiguration Video Input Connector [1..3] CameraControl Mode

Define whether the camera that is connected to this video input connector can be controlled or not.
Note that camera control is not available for Connector 2 (HDMI) and Connector 3 (VGA).

Requires user role: ADMIN
Default value: Connector 1: On  Connector 2,3: Off

USAGE:
```
xConfiguration Video Input Connector n CameraControl Mode: Mode
```
where
```
Mode: Connector 1: Off/On  Connector 2,3: Off
```
Off: Disable camera control.
On: Enable camera control.
xConfiguration Video Input Connector [1..3] CameraControl CameraId

The camera ID is a unique identifier of the cameras that are connected to the video input.

Requires user role: ADMIN
Default value: 1

**USAGE:**

```
xConfiguration Video Input Connector n CameraControl CameraId: CameraId
where
   CameraId: 1
```

The camera ID is fixed and cannot be changed.

xConfiguration Video Input Connector [1..3] InputSourceType

Select which type of input source is connected to the video input. Note that Connector 1 is the system’s integrated camera.

Requires user role: ADMIN
Default value: Connector 1: camera Other connectors: PC

**USAGE:**

```
xConfiguration Video Input Connector n InputSourceType: InputSourceType
where
   n: Index that identifies the input connector. Range: 1..3
   InputSourceType: Connector 1: camera Other connectors: camera/PC/mediaplayer/document_camera/whiteboard/other
```

- camera: Use this when a camera is connected to the video input.
- PC: Use this when a computer is connected to the video input.
- mediaplayer: Use this when a media player is connected to the video input.
- document_camera: Use this when a document camera is connected to the video input.
- whiteboard: Use this when a whiteboard camera is connected to the video input.
- other: Use this when the other options do not match.

xConfiguration Video Input Connector [1..3] Name

Define a name for the video input connector.

Requires user role: ADMIN
Default value: ""

**USAGE:**

```
xConfiguration Video Input Connector n Name: "Name"
where
   n: Index that identifies the input connector. Range: 1..3
   Name: String (0, 50)
```

Name for the video input connector.

xConfiguration Video Input Connector [2..3] Quality

When encoding and transmitting video there is a trade-off between high resolution and high frame rate. For some video sources it is more important to transmit high frame rate than high resolution and vice versa. This setting specifies whether to give priority to high frame rate or to high resolution.

Requires user role: ADMIN
Default value: Sharpness

**USAGE:**

```
xConfiguration Video Input Connector n Quality: Quality
where
   n: Index that identifies the input connector. Range: 2..3
   Quality: Motion/Sharpness
```

- Motion: Gives the highest possible frame rate. Used when there is a need for higher frame rates, typically when a large number of participants are present or when there is a lot of motion in the picture.
- Sharpness: Gives the highest possible resolution. Used when you want the highest quality of detailed images and graphics.
xConfiguration Video Input Connector [2..3] PresentationSelection

Define how the video system will behave when you connect a presentation source to the video input.

If the video system is in standby mode, it will wake up when you connect a presentation source.

Note that sharing the presentation with the far end always requires additional action (press Share on the user interface).

Requires user role: ADMIN
Default value: OnConnect

USAGE:
xConfiguration Video Input Connector \(n\) PresentationSelection:
\[\text{PresentationSelection}\]
where
\(n\): Index that identifies the input connector. Range: 2..3
\text{PresentationSelection}: Manual/OnConnect

- Manual: In manual mode, the contents of the video input will not be presented on the screen until you choose it from the user interface.
- OnConnect: When in on-connect mode, the content on the video input will be presented on screen when a cable is connected. Otherwise, the behavior is the same as in manual mode.

xConfiguration Video Input Connector [1..3] Visibility

Define the visibility of the video input connector in the menus on the user interface.

Note that Connector 1 is the system's integrated camera, which is not available as a presentation source.

> The default value is Always for Video Input Connector 2 Visibility (the HDMI connector).
> The default value is IfSignal for Video Input Connector 3 Visibility (the VGA connector).

Requires user role: ADMIN
Default value: Connector 1: Never  Connector 2: Always  Connector 3: OnConnect

USAGE:
xConfiguration Video Input Connector \(n\) Visibility: \(\text{Visibility}\)
where
\(\text{Visibility}\): Connector 1: Never  Connector 2, 3: Never/Always/IfSignal

- Never: When the input source is not expected to be used as a presentation source, set to Never.
- Always: When set to Always, the menu selection for the video input connector will always be visible on the graphical user interface.
- IfSignal: When set to IfSignal, the menu selection for the video input connector will only be visible when something is connected to the video input.

xConfiguration Video Monitors

Define the monitor layout mode. Note that this video system supports only one monitor, so this value is fixed and cannot be changed.

Requires user role: ADMIN
Default value: Single

USAGE:
xConfiguration Video Monitors: Monitors
where
\(\text{Monitors}\): Single

- Single: The layout is shown on the video system's monitor.
xConfiguration Video Output Connector [1] CEC Mode

This video output (HDMI) supports Consumer Electronics Control (CEC). When this setting is On (default is Off), the system will use CEC to set the monitor in standby when the system itself enters standby. Likewise the system will wake up the monitor when the system itself wakes up from standby. For this to happen, the monitor that is connected to the output must be CEC compatible and CEC must be configured on the monitor.

Note that the different manufacturers use different marketing names for CEC, for example Anynet+ (Samsung); Aquos Link (Sharp); BRAVIA Sync (Sony); HDMI-CEC (Hitachi); Kuro Link (Pioneer); CE-Link and Regza Link (Toshiba); RIHD (Onkyo); HDAVI Control, EZ-Sync, VIERA Link (Panasonic); EasyLink (Philips); and NetCommand for HDMI (Mitsubishi).

Requires user role: ADMIN
Default value: Off

**USAGE:**

```
xConfiguration Video Output Connector [1] CEC Mode: Mode
where
  Mode: Off/On
    Off: Disable CEC control
    On: Enable CEC control
```

xConfiguration Video Output Connector [1] OverscanLevel

Some monitors may not present the entire image that they receive. This means that the outer parts of the image that is sent from the video system may be cut off when displayed on the monitor.

Use this setting to instruct the video system not to use the outer part of the available frame. This part might be cut off by the monitor. Both the video and messages on screen will be scaled in this case.

Requires user role: ADMIN
Default value: None

**USAGE:**

```
xConfiguration Video Output Connector [1] OverscanLevel: OverscanLevel
where
  OverscanLevel: None/Medium/High
    None: The video system will use all of the output resolution.
    Medium: The video system will not use the outer 3% of the output resolution.
    High: The video system will not use the outer 6% of the output resolution.
```

xConfiguration Video Output Connector [1] Resolution

Define the resolution and refresh rate for the connected screen. This value is fixed and cannot be changed.

Default value: Auto

**USAGE:**

```
xConfiguration Video Output Connector [1] Resolution: Resolution
where
  Resolution: Auto
    Auto: The system will automatically try to set the optimal resolution based on negotiation with the connected monitor.
```
xConfiguration Video Presentation DefaultSource

Define which video input source to use as a default presentation source. This setting may be used by the API and 3rd party user interfaces, but is not relevant when using Touch 10 or the TRC6 remote control.

Requires user role: USER
Default value: 2

**USAGE:**

```
xConfiguration Video Presentation DefaultSource: DefaultSource
```

where

```
DefaultSource: 2
```

The video input source to use as default presentation source.

xConfiguration Video Selfview Default Mode

Define if the main video source (self-view) shall be displayed on screen after a call. The position and size of the self-view window is determined by the Video Selfview Default PIPPosition and the Video Selfview Default FullscreenMode settings respectively.

Requires user role: ADMIN
Default value: Current

**USAGE:**

```
xConfiguration Video Selfview Default Mode: Mode
```

where

```
Mode: Off/Current/On
```

Off: self-view is switched off when leaving a call.

Current: self-view is left as is, i.e. if it was on during the call, it remains on after the call; if it was off during the call, it remains off after the call.

On: self-view is switched on when leaving a call.

xConfiguration Video Selfview Default FullscreenMode

Define if the main video source (self-view) shall be shown in full screen or as a small picture-in-picture (PiP) after a call. The setting only takes effect when self-view is switched on (see the Video Selfview Default Mode setting).

Requires user role: ADMIN
Default value: Current

**USAGE:**

```
xConfiguration Video Selfview Default FullscreenMode: FullscreenMode
```

where

```
FullscreenMode: Off/Current/On
```

Off: self-view will be shown as a PiP.

Current: The size of the self-view picture will be kept unchanged when leaving a call, i.e. if it was a PiP during the call, it remains a PiP after the call; if it was fullscreen during the call, it remains fullscreen after the call.

On: The self-view picture will be shown in fullscreen.
xConfiguration Video Selfview Default PIPPosition

Define the position on screen of the small self-view picture-in-picture (PiP) after a call. The setting only takes effect when self-view is switched on (see the Video Selfview Default Mode setting) and fullscreen view is switched off (see the Video Selfview Default FullscreenMode setting).

Requires user role: ADMIN
Default value: Current

**USAGE:**

xConfiguration Video Selfview Default PIPPosition: \texttt{PIPPosition}

where

\texttt{PIPPosition}: Current/UpperLeft/UpperCenter/UpperRight/CenterLeft/CenterRight/
LowerLeft/LowerRight

- **Current**: The position of the self-view PiP will be kept unchanged when leaving a call.
- **UpperLeft**: The self-view PiP will appear in the upper left corner of the screen.
- **UpperCenter**: The self-view PiP will appear in the upper center position.
- **UpperRight**: The self-view PiP will appear in the upper right corner of the screen.
- **CenterLeft**: The self-view PiP will appear in the center left position.
- **CentreRight**: The self-view PiP will appear in the center right position.
- **LowerLeft**: The self-view PiP will appear in the lower left corner of the screen.
- **LowerRight**: The self-view PiP will appear in the lower right corner of the screen.

xConfiguration Video Selfview OnCall Mode

This setting is used to switch on self-view for a short while when setting up a call. The Video Selfview OnCall Duration setting determines for how long it remains on. This applies when self-view in general is switched off.

Requires user role: ADMIN
Default value: On

**USAGE:**

xConfiguration Video Selfview OnCall Mode: \texttt{Mode}

where

\texttt{Mode}: Off/On

- **Off**: self-view is not shown automatically during call setup.
- **On**: self-view is shown automatically during call setup.

xConfiguration Video Selfview OnCall Duration

This setting only has an effect when the Video Selfview OnCall Mode setting is switched On. In this case, the number of seconds set here determines for how long self-view is shown before it is automatically switched off.

Requires user role: ADMIN
Default value: 10

**USAGE:**

xConfiguration Video Selfview OnCall Duration: \texttt{Duration}

where

\texttt{Duration}: Integer (1..60)

- **Range**: Choose for how long self-view remains on. The valid range is between 1 and 60 seconds.
Chapter 4

xCommand commands
Description of the xCommand commands

In this chapter, you can find a complete list of all xCommand type commands with parameters.

We recommend you visit our web site regularly for updated versions of the manual.

Go to:  http://www.cisco.com/go/sx-docs

Audio commands .......................................................................................................................... 76
  xCommand Audio Microphones Mute ..................................................................................... 76
  xCommand Audio Microphones Unmute ................................................................................ 76
  xCommand Audio SoundsAndAlerts Ringtone List ............................................................... 76
  xCommand Audio SoundsAndAlerts Ringtone Play ............................................................. 76
  xCommand Audio Volume Decrease ..................................................................................... 76
  xCommand Audio Volume Increase ..................................................................................... 76
  xCommand Audio Volume Mute .......................................................................................... 76
  xCommand Audio Volume Set .......................................................................................... 77
  xCommand Audio Volume SetToDefault .......................................................................... 77
  xCommand Audio Volume Unmute ..................................................................................... 77

Bookings commands ................................................................................................................. 77
  xCommand Bookings Clear ................................................................................................. 77
  xCommand Bookings List ................................................................................................. 77

Call commands ......................................................................................................................... 78
  xCommand Call Accept ....................................................................................................... 78
  xCommand Call DTMFSend ............................................................................................... 78
  xCommand Call Disconnect ............................................................................................ 78
  xCommand Call FarEndControl Camera Move ................................................................. 78
  xCommand Call FarEndControl Camera Stop ................................................................. 79
  xCommand Call FarEndControl RoomPreset Activate ...................................................... 79
  xCommand Call FarEndControl Source Select .................................................................. 79
  xCommand Call Hold .......................................................................................................... 79

  xCommand Call Ignore ...................................................................................................... 80
  xCommand Call Join ......................................................................................................... 80
  xCommand Call Reject ..................................................................................................... 80
  xCommand Call Resume ................................................................................................... 80
  xCommand Call UnattendedTransfer .............................................................................. 80

CallHistory commands ............................................................................................................. 81
  xCommand CallHistory AcknowledgeAllMissedCalls ...................................................... 81
  xCommand CallHistory AcknowledgeMissedCall ............................................................. 81
  xCommand CallHistory DeleteAll ..................................................................................... 81
  xCommand CallHistory DeleteEntry ............................................................................... 81
  xCommand CallHistory Get ............................................................................................ 82
  xCommand CallHistory Recents ...................................................................................... 82

Camera commands .................................................................................................................. 83
  xCommand Camera PositionReset ................................................................................... 83
  xCommand Camera Preset Activate ................................................................................ 83
  xCommand Camera Preset ActivateDefaultPosition ..................................................... 83
  xCommand Camera Preset Edit ....................................................................................... 84
  xCommand Camera Preset List ....................................................................................... 84
  xCommand Camera Preset Remove .............................................................................. 84
  xCommand Camera Preset Show ..................................................................................... 85
  xCommand Camera Preset Store .................................................................................... 85
  xCommand Camera Ramp ................................................................................................ 86
  xCommand Camera TriggerAutofocus ............................................................................ 86

http://www.cisco.com/go/sx-docs
Conference commands .............................................. 87
xCommand Conference DoNotDisturb Activate .............. 87
xCommand Conference DoNotDisturb Deactivate ............. 87
xCommand Conference SpeakerLock Release .................. 87
xCommand Conference SpeakerLock Set ....................... 87
Diagnostics commands ............................................... 88
xCommand Diagnostics Run ........................................ 88
Dial commands ......................................................... 88
xCommand Dial ....................................................... 88
HttpFeedback commands ........................................... 89
xCommand HttpFeedback Deregister .............................. 89
xCommand HttpFeedback Register ................................. 89
Peripherals commands ............................................... 89
xCommand Peripherals Connect ................................. 89
xCommand Peripherals HeartBeat ............................... 90
xCommand Peripherals List ....................................... 90
xCommand Peripherals Pairing DeviceDiscovery Start .... 90
xCommand Peripherals Pairing Pair ............................ 90
xCommand Peripherals Pairing Unpair ......................... 91
xCommand Peripherals Purge ................................... 91
Phonebook commands ............................................. 91
xCommand Phonebook Contact Add ............................ 91
xCommand Phonebook Contact Delete ........................ 92
xCommand Phonebook Contact Modify ....................... 92
xCommand Phonebook ContactMethod Add .................. 93
xCommand Phonebook ContactMethod Delete ............... 93
xCommand Phonebook ContactMethod Modify ............... 94
xCommand Phonebook Folder Add ............................. 94
xCommand Phonebook Folder Delete .......................... 94
xCommand Phonebook Folder Modify ......................... 95
xCommand Phonebook Search .................................. 95
xCommand xCommand Phonebook Search ............................... 95
xCommand Conference DoNotDisturb Activate .......................... 87
xCommand Conference DoNotDisturb Deactivate ............. 87
xCommand Conference SpeakerLock Release .................. 87
xCommand Conference SpeakerLock Set ....................... 87
Diagnostics commands ............................................... 88
xCommand Diagnostics Run ........................................ 88
Dial commands ......................................................... 88
xCommand Dial ....................................................... 88
HttpFeedback commands ........................................... 89
xCommand HttpFeedback Deregister .............................. 89
xCommand HttpFeedback Register ................................. 89
Peripherals commands ............................................... 89
xCommand Peripherals Connect ................................. 89
xCommand Peripherals HeartBeat ............................... 90
xCommand Peripherals List ....................................... 90
xCommand Peripherals Pairing DeviceDiscovery Start .... 90
xCommand Peripherals Pairing Pair ............................ 90
xCommand Peripherals Pairing Unpair ......................... 91
xCommand Peripherals Purge ................................... 91
Phonebook commands ............................................. 91
xCommand Phonebook Contact Add ............................ 91
xCommand Phonebook Contact Delete ........................ 92
xCommand Phonebook Contact Modify ....................... 92
xCommand Phonebook ContactMethod Add .................. 93
xCommand Phonebook ContactMethod Delete ............... 93
xCommand Phonebook ContactMethod Modify ............... 94
xCommand Phonebook Folder Add ............................. 94
xCommand Phonebook Folder Delete .......................... 94
xCommand Phonebook Folder Modify ......................... 95
xCommand Phonebook Search .................................. 95
xCommand Presentation Start .................................. 96
xCommand Presentation Stop .................................... 96
Presentation commands ........................................... 96
xCommand Presentation Start .................................. 96
xCommand Presentation Stop .................................... 96
Provisioning commands .......................................... 97
xCommand Provisioning CompleteUpgrade ................... 97
xCommand Provisioning PostponeUpgrade .................... 97
xCommand Provisioning CUCM CTL Delete ................. 97
xCommand Provisioning CUCM CTL Show ................... 97
xCommand Provisioning StartUpgrade ......................... 97
xCommand Provisioning CUCM CTL Delete ................. 97
xCommand Provisioning CUCM CTL Show ................... 97
xCommand Provisioning StartUpgrade ......................... 97
xCommand Provisioning CUCM CTL Delete ................. 97
Provisioning commands .......................................... 97
xCommand Provisioning CompleteUpgrade ................... 97
xCommand Provisioning PostponeUpgrade .................... 97
xCommand Provisioning CUCM CTL Delete ................. 97
xCommand Provisioning CUCM CTL Show ................... 97
xCommand Provisioning StartUpgrade ......................... 97
xCommand Standby ResetTimer ................................ 100
xCommand Standby Deactivate .................................. 100
xCommand Standby ResetTimer ................................ 100
xCommand Standby Deactivate .................................. 100
xCommand Standby ResetTimer ................................ 100
SystemUnit commands .......................................... 100
xCommand SystemUnit Boot ................................... 100
xCommand SystemUnit FactoryReset ......................... 100
xCommand SystemUnit Notifications RemoveAll .......... 101
xCommand SystemUnit OptionKey Add ...................... 101
xCommand SystemUnit OptionKey Remove ................. 101
xCommand SystemUnit SoftwareUpgrade ................. 101
xCommand SystemUnit OptionKey Remove ................. 101
xCommand SystemUnit SoftwareUpgrade ................. 101
Time commands

- xCommand Time DateTime Get ................................................................. 102
- xCommand Time DateTime Set ................................................................. 102

UserInterface commands

- xCommand UserInterface Message Alert Clear ...................................... 102
- xCommand UserInterface Message Alert Display .................................... 102
- xCommand UserInterface Message Prompt Clear .................................... 103
- xCommand UserInterface Message Prompt Display ................................ 103
- xCommand UserInterface Message Prompt Response .............................. 104
- xCommand UserInterface Message TextLine Clear .................................. 104
- xCommand UserInterface Message TextLine Display ............................... 104
- xCommand UserInterface OSD Key Click .............................................. 104
- xCommand UserInterface OSD Key Press ............................................. 105
- xCommand UserInterface OSD Key Release ......................................... 105

UserManagement commands

- xCommand UserManagement RemoteSupportUser Create ....................... 105
- xCommand UserManagement RemoteSupportUser Delete ........................ 105
- xCommand UserManagement RemoteSupportUser DisablePermanently ...... 105
- xCommand UserManagement RemoteSupportUser GetState .................... 106
- xCommand UserManagement User Passphrase Change .......................... 106
- xCommand UserManagement User Passphrase Set ................................ 106

Video commands

- xCommand Video ActiveSpeakerPIP Set .................................................. 107
- xCommand Video Input Source SetActiveConnector ............................... 107
- xCommand Video Layout LayoutFamily Set ............................................ 107
- xCommand Video PresentationPIP Set .................................................... 107
- xCommand Video Selfview Set ......................................................... 108
Audio commands

xCommand Audio Microphones Mute
Mute all microphones.
Requires user role: USER

USAGE:
xCommand Audio Microphones Mute

xCommand Audio Microphones Unmute
Unmute microphones.
Requires user role: USER

USAGE:
xCommand Audio Microphones Unmute

xCommand Audio SoundsAndAlerts Ringtone List
Lists all available ringtones that can be configured using xConfiguration Audio SoundsAndAlerts RingTone.
Requires user role: USER

USAGE:
xCommand Audio SoundsAndAlerts Ringtone List

xCommand Audio SoundsAndAlerts Ringtone Play
Play one of the available ringtones. To get a list of the available ringtones use the command xCommand Audio SoundsAndAlerts Ringtone List.
Requires user role: USER

USAGE:
xCommand Audio SoundsAndAlerts Ringtone Play RingTone: "RingTone"
where
RingTone: String (1, 100)
The name of the ringtone.

xCommand Audio Volume Decrease
Decrease the volume on the endpoint.
Requires user role: USER

USAGE:
xCommand Audio Volume Decrease [Steps: Steps]
where
Steps: Integer (1..10)
One step equals 0.5dB decrease in volume.

xCommand Audio Volume Increase
Increase the volume on the endpoint.
Requires user role: USER

USAGE:
xCommand Audio Volume Increase [Steps: Steps]
where
Steps: Integer (1..10)
One step equals 0.5dB increase in volume.

xCommand Audio Volume Mute
Mute the volume on the endpoint.
Requires user role: USER

USAGE:
xCommand Audio Volume Mute
xCommand Audio Volume Set
Set the volume on the endpoint to a specified level.
Requires user role: USER

**USAGE:**
xCommand Audio Volume Set Level: Level
where
Level: Integer (0..100)
Select gain level. The default level 70 equals 0dB gain. Level 100 equals 15db gain.

xCommand Audio Volume SetToDefault
Set the current volume level as the default for the endpoint.
Requires user role: USER

**USAGE:**
xCommand Audio Volume SetToDefault

xCommand Audio Volume Unmute
Set the volume on the endpoint back on after muting.
Requires user role: USER

**USAGE:**
xCommand Audio Volume Unmute

---

**Bookings commands**

xCommand Bookings Clear
Clear the current stored list of bookings.
Requires user role: USER

**USAGE:**
xCommand Bookings Clear

xCommand Bookings List
List the stored bookings for the system. The list of booking details is received from the management system. All parameters are optional, and can be used to limit the search result. If no parameters are set, past, present and future bookings are all listed. To avoid listing bookings from yesterday and before, use DayOffset = 0.
Requires user role: USER

**USAGE:**
xCommand Bookings List [Days: Days] [DayOffset: DayOffset] [Limit: Limit] [Offset: Offset]
where
Days: Integer (1..365)
Number of days to retrieve bookings from.
DayOffset: Integer (0..365)
Which day to start the search from (today: 0, tomorrow: 1..).
Limit: Integer (1..65534)
Max number of bookings to list.
Offset: Integer (0..65534)
Offset number of bookings for this search.
Call commands

xCommand Call Accept
Accept an incoming call. If no CallId is specified, all incoming calls are accepted.
Requires user role: USER

USAGE:
xCommand Call Accept [CallId: CallId]
where
CallId: Integer (0..65534)
The CallID is returned when the xCommand Dial command is run. During the call you can run the xStatus Call command to see the CallId.

xCommand Call DTMFSend
Send DTMF tones to the far end.
Requires user role: USER

USAGE:
xCommand Call DTMFSend [CallId: CallId] DTMFString: "DTMFString"
where
CallId: Integer (0..65534)
The CallID is returned when the xCommand Dial command is run. During the call you can run the xStatus Call command to see the CallId.

DTMFString: String (0, 32)
Enter the DTMF string.

xCommand Call Disconnect
Disconnect a call.
Requires user role: USER

USAGE:
xCommand Call Disconnect [CallId: CallId]
where
CallId: Integer (0..65534)
The CallID is returned when the xCommand Dial command is run. During the call you can run the xStatus Call command to see the CallId.

xCommand Call FarEndControl Camera Move
Move the far end camera (the remote camera).
NOTE: The far end camera moves in the specified direction until the stop command (ref: xCommand FarEndControl Camera Stop) is issued.
Requires user role: USER

USAGE:
xCommand Call FarEndControl Camera Move [CallId: CallId] Value: Value
where
CallId: Integer (0..65534)
The CallID is returned when the xCommand Dial command is run. During the call you can run the xStatus Call command to see the CallId.

Value: Left/Right/Up/Down/ZoomIn/ZoomOut
Select the action for how to move the camera.
xCommand Call FarEndControl Camera Stop
Stop the far end camera after the xCommand FarEndControl Camera Move has been issued.
Requires user role: USER

USAGE:
```
xCommand Call FarEndControl Camera Stop [CallId: CallId]
```

CallId: Integer (0..65534)
The CallID is returned when the xCommand Dial command is run. During a call you can run the xStatus Call command to see the CallId.

xCommand Call FarEndControl RoomPreset Activate
While in a call, this command is used to activate a preset on the far end codec. The preset covers the far end codec’s camera positions and input video switcher settings.
The preset must be stored on the far end codec beforehand, either by using the xCommand Preset Store command locally on the far end codec, or by using the xCommand FarEndControl Preset Store command from a remote codec.
Note: The far end codec’s xConfiguration Conference FarEndControl Mode setting must be switched On for the FarEndControl commands to work.
Requires user role: USER

USAGE:
```
xCommand Call FarEndControl RoomPreset Activate [CallId: CallId] PresetId: PresetId
```

CallId: Integer (0..65534)
The CallId is returned when issuing the xCommand Dial command. During a call you can run the xStatus Call command to see the CallId.

PresetId: Integer (1..15)
The ID of the preset that is stored on the far end codec.

xCommand Call FarEndControl Source Select
Select which video input source to use as the main source on the far end system.
Requires user role: USER

USAGE:
```
xCommand Call FarEndControl Source Select [CallId: CallId] SourceId: SourceId
```

CallId: Integer (0..65534)
The CallID is returned when the xCommand Dial command is run. During a call you can run the xStatus Call command to see the CallId.

SourceId: Integer (1..15)
Select a video input source on the far end.

xCommand Call Hold
Put a call on hold.
Requires user role: USER

USAGE:
```
xCommand Call Hold [CallId: CallId] [Reason: Reason]
```

CallId: Integer (0..65534)
The CallID is returned when the xCommand Dial command is run. During a call you can run the xStatus Call command to see the CallId.

Reason: Conference/Transfer/Other
Internal usage only.
xCommand Call Ignore

Turns off the ringtone for the incoming call. The call can still be answered.

Requires user role: USER

**USAGE:**

xCommand Call Ignore CallId: **CallId**

where

**CallId**: Integer (0..65534)

The CallID is returned when the xCommand Dial command is run. During a call you can run the xStatus Call command to see the CallId.

xCommand Call Join

Internal usage only.

Requires user role: USER

**USAGE:**

xCommand Call Join CallId: **CallId**

where

**CallId**: Integer (0..65534)

xCommand Call Reject

Reject incoming call. If no call id is specified, all incoming calls are rejected.

Requires user role: USER

**USAGE:**

xCommand Call Reject [CallId: **CallId**]

where

**CallId**: Integer (0..65534)

The CallID is returned when the xCommand Dial command is run. During a call you can run the xStatus Call command to see the CallId.

xCommand Call Resume

Resume a call that have been put on hold.

Requires user role: USER

**USAGE:**

xCommand Call Resume CallId: **CallId**

where

**CallId**: Integer (0..65534)

The CallID is returned when the xCommand Dial command is run. During a call you can run the xStatus Call command to see the CallId.

xCommand Call UnattendedTransfer

Transfers an ongoing call to another participant. Fully supported for SIP calls only.

Requires user role: USER

**USAGE:**

xCommand Call UnattendedTransfer CallId: **CallId** Number: "**Number**"

where

**CallId**: Integer (0..65534)

The CallID is returned when the xCommand Dial command is run. During a call you can run the xStatus Call command to see the CallId.

**Number**: String (0, 255)

The number the call is transfered to.
CallHistory commands

xCommand CallHistory AcknowledgeAllMissedCalls
Turns off the missed calls indicator on the touch controller for all missed calls.
Requires user role: USER

USAGE:
   xCommand CallHistory AcknowledgeAllMissedCalls

xCommand CallHistory AcknowledgeMissedCall
Turns off the missed calls indicator on the touch controller for the specified call.
Requires user role: USER

USAGE:
   xCommand CallHistory AcknowledgeMissedCall CallHistoryId: CallHistoryId
   [AcknowledgeConsecutiveDuplicates: AcknowledgeConsecutiveDuplicates]
   where
   CallHistoryId: Integer (1..2147483647)
   CallHistoryId for the call in question. Run xCommand CallHistory Get to get the id number.
   AcknowledgeConsecutiveDuplicates: False/True
   You can include or exclude all surrounding calls with duplicate information.

xCommand CallHistory DeleteEntry
Deletes all information on the specified call.
Requires user role: USER

USAGE:
   xCommand CallHistory DeleteEntry CallHistoryId: CallHistoryId
   [AcknowledgeConsecutiveDuplicates: AcknowledgeConsecutiveDuplicates]
   where
   CallHistoryId: Integer (1..2147483647)
   CallHistoryId for the call in question. Run xCommand CallHistory Get to get the id number.
   AcknowledgeConsecutiveDuplicates: False/True
   You can include or exclude all surrounding calls with duplicate information.

xCommand CallHistory DeleteAll
Deletes all information on previous calls.
Requires user role: USER

USAGE:
   xCommand CallHistory DeleteAll [Filter: Filter]
   where
   Filter: All/Missed/Placed/Received
   You can filter which calls to delete.
xCommand CallHistory Get
Retrieve all information on previous calls made on the video system.
Requires user role: USER

USAGE:
xCommand CallHistory Get [Filter: Filter] [Offset: Offset] [Limit: Limit]
[DetailLevel: DetailLevel] [SearchString: "SearchString"] [CallHistoryId: CallHistoryId]
where
Filter: All/Missed/AnsweredElsewhere/Forwarded/Placed/NoAnswer/Received/Rejected/UnacknowledgedMissed
You can filter which calls to retrieve.

Offset: Integer (0..65534)
Sets the call from which to start.

Limit: Integer (0..65534)
Defines the amount of calls in the output.

DetailLevel: Basic/Full
Sets the level of detail for the information on these calls.

SearchString: String ( 0, 255)
Allows you to set the command to apply to a specified display name or call back number.

CallHistoryId: Integer (0..65534)
CallHistoryId for the call in question.

xCommand CallHistory Recents
Retrieve aggregated information on previous calls made on the video system.
Requires user role: USER

USAGE:
xCommand CallHistory Recents [Filter: Filter] [Offset: Offset] [Limit: Limit]
[DetailLevel: DetailLevel] [SearchString: "SearchString"] [CallHistoryId: CallHistoryId]
[Order: Order]
where
Filter: All/Missed/AnsweredElsewhere/Forwarded/Placed/NoAnswer/Received/Rejected/UnacknowledgedMissed
You can filter which calls to retrieve.

Offset: Integer (0..65534)
Sets the call from which to start.

Limit: Integer (0..65534)
Defines the amount of calls in the output.

DetailLevel: Basic/Full
Sets the level of detail for the information on these calls.

SearchString: String ( 0, 255)
Allows you to set the command to apply to a specified display name or call back number.

CallHistoryId: Integer (0..65534)
CallHistoryId for the call in question.

Order: OccurrenceTime/OccurrenceFrequency
Define the order in which the previous calls are presented.
Camera commands

**xCommand Camera PositionReset**
Reset the camera position to default position.
Requires user role: USER

**USAGE:**
xCommand Camera PositionReset [Axis: **Axis**] CameraId: **CameraId**

where

**Axis:** All/Focus/PanTilt/Zoom
Select which motor to reset. If not specified all are reset.

**CameraId:** Integer (1..1)
The ID of the camera preset you want to reset.

**xCommand Camera Preset Activate**
Activate one of the stored camera presets.
Note that the xCommand Camera Preset commands applies to an individual camera. This is in contrast to the xCommand Preset commands where a single preset covers ALL connected cameras plus the Video Input switcher settings.
Requires user role: USER

**USAGE:**
xCommand Camera Preset Activate PresetId: **PresetId**

where

**PresetId:** Integer (1..35)
The ID of the camera preset you want to activate.

**xCommand Camera Preset ActivateDefaultPosition**
Sets the cameras to their default position, if one is defined. The default position is defined by xCommand Camera Preset Store or by xCommand Camera Preset Edit. Only one default position can be defined per camera.
Requires user role: USER

**USAGE:**
xCommand Camera Preset ActivateDefaultPosition [CameraId: **CameraId**]

where

**CameraId:** Integer (1..1)
The ID of the camera preset you want to activate. If CameraId is not specified, all cameras will be set in their respective default position, if one is defined.
xCommand Camera Preset Edit

Edit a stored camera preset. You can change the name of the camera preset and its position in the list that is returned by the xCommand Camera Preset List command. You can also change whether or not this preset is the default position for the associated camera. Note that the xCommand Camera Preset commands applies to an individual camera. This is in contrast to the xCommand Preset commands where a single preset covers ALL connected cameras and the Video Input switcher settings.

Requires user role: USER

USAGE:

xCommand Camera Preset Edit PresetId: PresetId [ListPosition: ListPosition] [Name: "Name"] [DefaultProsition: DefaultProsition]

where

PresetId: Integer (1..35)  
The ID of the camera preset you want to edit.

ListPosition: Integer (1..35)  
The position in the list returned by the xCommand Camera Preset List command.

Name: String (0, 255)  
The name of the camera preset. It will be used in the list returned by the xCommand Camera Preset List command.

DefaultProsition: False/True  
Defines whether or not this preset is the default position for the associated camera. Note that each camera can only have one default position, so if set, the old default preset will automatically be marked as not default.

xCommand Camera Preset List

List information about available camera presets. Note that the xCommand Camera Preset commands applies to an individual camera. This is in contrast to the xCommand Preset commands where a single preset covers ALL connected cameras plus the Video Input switcher settings.

Requires user role: USER

USAGE:

xCommand Camera Preset List CameraId: CameraId [DefaultPosition: DefaultPosition]

where

CameraId: Integer (1..1)  
Only list presets for the specified camera.

DefaultPosition: False/True  
List default positions only, or only those that are not default positions.

xCommand Camera Preset Remove

Remove a camera preset. Note that the xCommand Camera Preset commands applies to an individual camera. This is in contrast to the xCommand Preset commands where a single preset covers ALL connected cameras plus the Video Input switcher settings.

Requires user role: USER

USAGE:

xCommand Camera Preset Remove PresetId: PresetId

where

PresetId: Integer (1..35)  
The ID of the camera preset you want to remove.
xCommand Camera Preset Show

Shows the preset details for the requested PresetId.

Requires user role: USER

**USAGE:**

xCommand Camera Preset Show PresetId: **PresetId**

where

**PresetId:** Integer (1..35)

  The ID of the camera preset you wish to see.

xCommand Camera Preset Store

Store the current position (pan and tilt), zoom and focus of the chosen camera. The camera is identified by the CameraId parameter.

Note that the xCommand Camera Preset commands applies to an individual camera. This is in contrast to the xCommand Preset commands where a single preset covers ALL connected cameras plus the Video Input switcher settings. The xCommand Camera Preset commands are useful when you want to handle multiple camera positions individually per camera, rather than working with complete sets of camera positions. The individual camera presets are not available for far end control.

Requires user role: USER

**USAGE:**

xCommand Camera Preset Store [PresetId: **PresetId**] CameraId: **CameraId**

[ListPosition: **ListPosition**] [Name: "**Name**"] [TakeSnapshot: **TakeSnapshot**]

[DefaultProsition: **DefaultProsition**]

where

**PresetId:** Integer (1..35)

  The ID of this camera preset. If not set explicitly, the codec will assign a preset ID automatically.

**CameraId:** Integer (1..1)

  Select the camera for which to store the preset position.

**ListPosition:** Integer (1..35)

  The new camera preset's position in the list returned by the xCommand Camera Preset List command.

**Name:** String (0, 255)

  The name of the new camera preset. It will be used in the list returned by the xCommand Camera Preset List command.

**TakeSnapshot:** False/True

  Allow or disallow snapshot of the preview.

**DefaultProsition:** False/True

  Defines whether or not this preset shall be the default position of the associated camera. Note that each camera can hold only one default position, so if set, the old default preset will automatically be marked as not default.
xCommand Camera Ramp

Move the camera in a specified direction. The camera moves at specified speed until a stop command is issued. In a daisy chain, you need to know the CameraId for the camera you want to address. Be aware that pan and tilt can be operated simultaneously, but no other combinations. In the latter case only the first operation specified is executed. For example, if you try to run both zoom and pan at the same time, only zoom is executed.

NOTE: You must run a stop command to stop the camera, see the example below.

Requires user role: USER

USAGE:

```
xCommand Camera Ramp CameraId: CameraId [Pan: Pan] [PanSpeed: PanSpeed] [Tilt: Tilt] [TiltSpeed: TiltSpeed] [Zoom: Zoom] [ZoomSpeed: ZoomSpeed] [Focus: Focus]
```

where

**CameraId:** Integer (1..1)  
Select the camera.

**Pan:** Left/Right/Stop  
Move the camera to the Left or Right, followed by Stop.

**PanSpeed:** Integer (1..15)  
Set the pan speed.

**Tilt:** Down/Up/Stop  
Move the camera Up or Down, followed by Stop.

**TiltSpeed:** Integer (1..15)  
Set the tilt speed.

**Zoom:** In/Out/Stop  
Zoom the camera In or Out, followed by Stop.

**ZoomSpeed:** Integer (1..15)  
Set the zoom speed.

**Focus:** Far/Near/Stop  
Focus the camera Far or Near, followed by Stop.

xCommand Camera TriggerAutofocus

Trigger the auto-focus functionality. The camera must support auto-focus functionality. If the camera is daisy chained, the CameraId is given by its place in the chain.

Requires user role: USER

USAGE:

```
xCommand Camera TriggerAutofocus CameraId: CameraId
```

where

**CameraId:** Integer (1..1)  
Select the camera to auto-focus.
Conference commands

**xCommand Conference DoNotDisturb Activate**
This command switches on the Do Not Disturb mode, and the Timeout parameter allows you to control when it is switched off again. When Do Not Disturb is switched on, all incoming calls are rejected and registered as missed calls. The calling side receives a busy signal.

Requires user role: USER

**USAGE:**
```
xCommand Conference DoNotDisturb Activate [Timeout: Timeout]
```

Timeout: Integer (0..1440)
Set the number of minutes before Do Not Disturb is switched off. If not set, Do Not Disturb times out after 1440 minutes (24 hours).

**xCommand Conference DoNotDisturb Deactivate**
Switch off the Do Not Disturb mode. When Do Not Disturb is switched off incoming calls come through as normal.

Requires user role: USER

**USAGE:**
```
xCommand Conference DoNotDisturb Deactivate
```

**xCommand Conference SpeakerLock Release**
Releases locked speaker set by xCommand Conference SpeakerLock Set. Default voice switching is switched back on.

Requires user role: USER

**USAGE:**
```
xCommand Conference SpeakerLock Release
```

**xCommand Conference SpeakerLock Set**
For manually locking one of the speakers to the prominent speaker position. This overrides the default voice switching.

Requires user role: USER

**USAGE:**
```
xCommand Conference SpeakerLock Set Target: Target [CallId: CallId]
```

Target: local/remote
Identifies local or remote participant.

CallId: Integer (0..65534)
Identify CallID for the remote participant. Only relevant if Target is set to "remote".
Diagnostics commands

xCommand Diagnostics Run
This command runs self-diagnostics commands on the system.
Requires user role: ADMIN

**USAGE:**
xCommand Diagnostics Run [ResultSet: ResultSet]

**ResultSet:** Alerts/All/None
You can filter the diagnostics results to alerts, all or none. If not set, the result will show all results.

Dial commands

xCommand Dial
Dial out from the system. Returns information about the CallId and ConferenceId, which are required for some of the other commands.
Requires user role: USER

**USAGE:**
xCommand Dial Number: "Number" [Protocol: Protocol] [CallRate: CallRate] [CallType: CallType] [BookingId: "BookingId"] [Appearance: Appearance] [DisplayName: "DisplayName"]

**Number:** String (0, 255)
Enter the number or address.

**Protocol:** H320/Sip
Select the call protocol.

**CallRate:** Integer (64..3072)
Set the call rate.

**CallType:** Audio/Video
Select the call type.

**BookingId:** String (0, 255)
Any identifier that an external booking system (e.g. TMS, CTS-MAN) can use for its own references to match placed calls with the booking systems internal identifier for a meeting. This can be any string, e.g. a GUID. The booking Id is supplied in call logs, call events etc for the call.

**Appearance:** Integer (1..999999999)
Internal usage only.

**DisplayName:** String (0, 255)
The display name of the remote participant.
HttpFeedback commands

xCommand HttpFeedback Deregister
Deregister XML feedback over HTTP(S).
Requires user role: ADMIN

USAGE:
xCommand HttpFeedback Deregister FeedbackSlot: FeedbackSlot
where
FeedbackSlot: Integer (1..4)
You can have from 1 to 4 slots for feedback.

xCommand HttpFeedback Register
Register the system to a HTTP(S) server to return XML feedback over HTTP(S) to specific URLs.
Requires user role: ADMIN

USAGE:
xCommand HttpFeedback Register [FeedbackSlot: FeedbackSlot] ServerUrl: ServerUrl
where
FeedbackSlot: Integer (1..4)
You can have from 1 to 4 slots for feedback.
ServerUrl: String (1, 2048)
Define the URL for the HTTP(S) server.
Expression[1..15]: String (1, 255)
XPath expressions specify which parts of the Status and Configuration XML documents are monitored. You can have from 1 to 15 XPath expressions.

Peripherals commands

xCommand Peripherals Connect
Register peripherals that are connected to the codec, such as control systems and touch panels. The registered peripherals are displayed on the web interface under Configuration > Peripherals.
This command should be used when the peripheral connects to the codec for the first time or when the software version on the peripheral has changed. The list of connected devices is available with the command xStatus Peripherals ConnectedDevice [n] Status.
Requires user role: USER

USAGE:
xCommand Peripherals Connect [HardwareInfo: "HardwareInfo"] ID: "ID" [Name: "Name"] [NetworkAddress: "NetworkAddress"] [SerialNumber: "SerialNumber"] [SoftwareInfo: "SoftwareInfo"] Type: Type
where
HardwareInfo: String (0, 100)
The device’s hardware number.
ID: String (1, 100)
A unique ID for the device you are connecting to, typically a MAC address.
Name: String (0, 100)
Define a name for the device.
NetworkAddress: String (0, 100)
Network address for the device you are connecting to.
SerialNumber: String (0, 100)
The device’s serial number.
SoftwareInfo: String (0, 100)
Software version the device is running.
Type: Byod/ControlSystem/Other/TouchPanel
Define the type of device you are connecting to.
xCommand Peripherals HeartBeat
When a peripheral is registered as a connected device, you can set it to send a heartbeat to the codec to let the codec know that it is still connected. This will keep the device on the xStatus Peripherals ConnectedDevice list. If the peripheral is not set to send a heartbeat, the device will disappear from the list after a while.
Note: Does not apply to cameras.
Requires user role: USER

USAGE:
```plaintext
xCommand Peripherals HeartBeat ID: "ID" [Timeout: Timeout]
```
where
- **ID**: String (1, 100)
  A unique ID for the device you are connecting to, typically a MAC address.
- **Timeout**: Integer (1..65535)
  Set how long the device will send heartbeat.

xCommand Peripherals List
Lists all currently and previously connected peripherals.
Requires user role: ADMIN

USAGE:
```plaintext
xCommand Peripherals List [Connected: Connected] [Type: Type]
```
where
- **Connected**: False/True
  Limit the search to currently connected devices.
- **Type**: All/ControlSystem/ISDNLink/Other/TouchPanel
  Limit the search by device type.

xCommand Peripherals Pairing DeviceDiscovery Start
Start device discovery to detect ISDN Links in the same network.
Requires user role: ADMIN

USAGE:
```plaintext
xCommand Peripherals Pairing DeviceDiscovery Start [AutoPairing: AutoPairing] [DeviceType: DeviceType] [Timeout: Timeout]
```
where
- **AutoPairing**: On/Off
  You can select to automatically pair the detected device to the endpoint.
- **DeviceType**: ISDNLink
  Only look for ISDN Link.
- **Timeout**: Integer (3..60)
  Set a maximum time for the search from 3 to 60 seconds.

xCommand Peripherals Pairing Pair
Pair an ISDN Link to an endpoint.
Requires user role: ADMIN

USAGE:
```plaintext
xCommand Peripherals Pairing Pair MacAddress: "MacAddress"
```
where
- **MacAddress**: String (1, 1450)
  Enter the MAC address for the ISDN Link you wish to pair to the endpoint.
xCommand Peripherals Pairing Unpair

Unpair endpoint from an ISDN Link, when the two have contact.

Requires user role: ADMIN

**USAGE:**

```plaintext
xCommand Peripherals Pairing Unpair MacAddress: "MacAddress"
```

Where

**MacAddress:** String (1, 100)

MacAddress: Enter the MAC address for the ISDN Link you wish to unpair from the endpoint.

xCommand Peripherals Purge

Force unpair an endpoint from an ISDN Link when a connection has been lost. Note: You must also unpair the ISDN Link to be able to pair it to another endpoint.

Requires user role: USER

**USAGE:**

```plaintext
xCommand Peripherals Purge ID: "ID"
```

Where

**ID:** String (1, 100)

ID: Mac address of the ISDN Link in the format "xx:xx:xx:xx:xx:xx".

Phonebook commands

xCommand Phonebook Contact Add

Requires user role: USER

**USAGE:**

```plaintext
xCommand Phonebook Contact Add Name: "Name" [FolderId: "FolderId"] [ImageURL: "ImageURL"] [Title: "Title"] [Number: "Number"] [Protocol: Protocol] [CallRate: CallRate] [CallType: CallType] [Device: Device] [Tag: Tag]
```

Where

**Name:** String (0, 255)

Name: The name of the contact.

**FolderId:** String (0, 255)

FolderId: The unique identifier for the folder that you want to store the contact in. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook Folder Add command was issued to make the folder.

**ImageURL:** String (0, 255)

ImageURL: Currently not in use.

**Title:** String (0, 255)

Title: The title of the contact.

**Number:** String (0, 255)

Number: The phone number or address of the contact.

**Protocol:** Auto/H320/H323/SIP

Protocol: Select the Auto, SIP, H323 or H320 protocol.

**CallRate:** Integer (0..6000)

CallRate: Set a call rate.

**CallType:** Audio/Video

CallType: Select a call type (audio or video).

**Device:** Mobile/Other/Telephone/Video

Device: Select the device type.
Tag: Untagged/Favorite
Tag the contact as a Favorite, or untag an already tagged contact.

xCommand Phonebook Contact Delete
Delete an existing contact from the local phonebook.
Requires user role: USER

USAGE:
xCommand Phonebook Contact Delete ContactId: "ContactId"
where
ContactId: String (0, 255)
The unique identifier for the contact. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook Contact Add command was issued to make the contact.

xCommand Phonebook Contact Modify
Modify contact details of an existing contact in the local phonebook. The following parameters can be changed using this command: Name, FolderId, ImageURL and Title. You must use the xCommand Phonebook ContactMethod Modify command to change the other parameters: Number, Protocol, CallRate, CallType and Device.
Requires user role: USER

USAGE:
xCommand Phonebook Contact Modify ContactId: "ContactId" [Name: "Name"] [FolderId: "FolderId"] [ImageURL: "ImageURL"] [Title: "Title"] [Tag: Tag]
where
ContactId: String (0, 255)
The unique identifier for the contact you want to modify. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook Contact Add command was issued to make the contact.

Name: String (0, 255)
The name of the contact.

FolderId: String (0, 255)
A unique identifier for the folder. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook Folder Add command was issued.

ImageURL: String (0, 255)
Currently not in use.

Title: String (0, 255)
The title of the contact.

Tag: Untagged/Favorite
Tag the contact as a Favorite, or untag an already tagged contact.
xCommand Phonebook ContactMethod Add

Add contact details for an existing contact in the local phonebook. The command returns the ContactMethodId, which is a unique string that identifies the contact method, typically the format is "n".

You can add several contact methods to a contact. Note that only the first contact method will appear in the Favorites list on the Cisco TelePresence Touch controller. The first contact method may have been created when issuing the xCommand Phonebook Contact Add command to make the contact. All contact methods are available in the API, on the web interface and when using the remote control.

Requires user role: USER

USAGE:

```
xCommand Phonebook ContactMethod Add ContactId: "ContactId" [Device: Device] Number: "Number" [Protocol: Protocol] [CallRate: CallRate] [CallType: CallType]
```

where

- **ContactId**: String (0, 255)
  The unique identifier for the contact that you want to add a contact method to. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook Contact Add command was issued to make the contact.

- **Device**: Mobile/Other/Telephone/Video
  Set which type of device to call to.

- **Number**: String (0, 255)
  The phone number or address of the contact.

- **Protocol**: Auto/H320/H323/SIP
  Select Auto, SIP, H323 or H320 protocol.

- **CallRate**: Integer (0..6000)
  Set a call rate.

- **CallType**: Audio/Video
  Select a call type (audio or video).

xCommand Phonebook ContactMethod Delete

Delete a contact method from an existing contact in the local phonebook.

Requires user role: USER

USAGE:

```
xCommand Phonebook ContactMethod Delete ContactId: "ContactId" ContactMethodId: "ContactMethodId"
```

where

- **ContactId**: String (0, 255)
  The unique identifier for the contact you want to change. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook Contact Add command was issued to make the contact.

- **ContactMethodId**: String (0, 255)
  The unique identifier for the contact method you want to delete. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook ContactMethod Add command was issued to make the contact method.
xCommand Phonebook ContactMethod Modify
Modify details about the contact method for an existing contact in the local phonebook.
Requires user role: USER

USAGE:
xCommand Phonebook ContactMethod Modify ContactId: "ContactId" ContactMethodId: "ContactMethodId" [Device: Device] [Number: "Number"] [Protocol: Protocol] [CallRate: CallRate] [CallType: CallType]
where
ContactId: String (0, 255)
The unique identifier for the contact. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook Contact Add command was issued to make the contact.

ContactMethodId: String (0, 255)
The unique identifier for the contact method you want to modify. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook ContactMethod Add or xCommand Phonebook Contact Add commands were issued to make the contact.

Device: Mobile/Other/Telephone/Video
Set which type of device to call to.

Number: String (0, 255)
The phone number or address of the contact.

Protocol: Auto/H320/H323/SIP
Select Auto, SIP, H323 or H320 protocol.

CallRate: Integer (0..6000)
Set a call rate.

CallType: Audio/Video
Select a call type (audio or video).

xCommand Phonebook Folder Add
Phonebook entries can be stored in folders. Use this command to add a folder to the local phonebook. The command returns the FolderId, which is a unique string that identifies the folder; typically the format is “localGroupId-n”.
Requires user role: USER

USAGE:
xCommand Phonebook Folder Add Name: "Name" [ParentFolderId: "ParentFolderId"]
where
Name: String (0, 255)
The name of the folder.

ParentFolderId: String (0, 255)
The unique identifier for the parent folder. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook Folder Add command was issued to make the parent folder.

xCommand Phonebook Folder Delete
Delete an existing folder from the local phonebook.
Requires user role: USER

USAGE:
xCommand Phonebook Folder Delete FolderId: "FolderId"
where
FolderId: String (0, 255)
The unique identifier for the folder. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook Folder Add command was issued to make the folder.
xCommand Phonebook Folder Modify

Modify an existing phonebook folder.

Requires user role: USER

**USAGE:**

xCommand Phonebook Folder Modify FolderId: "FolderId" [Name: "Name"] [ParentFolderId: "ParentFolderId"]

Where

FolderId: String (0, 255)
The unique identifier for the folder. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook Folder Add command was issued to make the folder.

Name: String (0, 255)
The name of the contact.

ParentFolderId: String (0, 255)
The unique identifier for the parent folder. The identifier will be returned by an xCommand Phonebook Search command. It was also returned when the xCommand Phonebook Folder Add command was issued to make the parent folder.

xCommand Phonebook Search

The search command lets you search in both the local and corporate phone books. A search gives a ResultSet.

The total number of folders and contacts (TotalRows) is always included in the result set when searching the local phone book. When searching a corporate phonebook the total number of folders and contacts may not be included. Whether it is included or not depends on the backend corporate phonebook service (e.g. CUCM, VCS, TMS) and its version.

Requires user role: USER

**USAGE:**

xCommand Phonebook Search [PhonebookId: "PhonebookId"] [PhonebookType: Corporate/Local] [SearchString: "SearchString"] [SearchField: Name/Number] [Offset: Offset] [FolderId: "FolderId"] [Limit: Limit] [Recursive: Recursive] [ContactType: ContactType] [Tag: Tag]

Where

PhonebookId: String (0, 255)
The identifier of the phonebook server that will be searched. See the xConfiguration Phonebook Server ID setting.

PhonebookType: Corporate/Local
Define whether to search the local phone book or the corporate phonebook.

SearchString: String (0, 255)
Search for entries containing this string (note that the entry does not have to begin with the string). If no FolderId is specified, all folders / phonebook directories will be searched.

SearchField: Name/Number
Currently not in use.

Offset: Integer (0..65534)
Get records starting with this offset in a search. The default value is 0. Offset is used together with Limit to support paging.

FolderId: String (0, 255)
Search only in the specified folder. The FolderId (string) is listed in the ResultSet of a search result containing folders.

Limit: Integer (0..65534)
Limit the number of records in the result set to this number. For example, if the limit is
set to 10, the ResultSet will contain only 10 entries (Contacts and Folders) even if the total number of hits is larger. The maximum limit is 1000.

**Recursive:** False/True

This parameter will only have effect when searching the local phone book. The setting determines whether a local phone book search should be limited to the given FolderId, or also recursively search in its subfolders. If not specified, the search will be recursive.

When issuing the command without specifying any parameters, all folders, contacts and contact methods in the local phone book will be returned.

**ContactType:** Any/Folder/Contact

Search all contact types, or limit the search to folders or individual contacts.

**Tag:** Untagged/Favorite

Limits the search to either contacts that have been tagged as favorite or the untagged contacts.

---

**Presentation commands**

**xCommand Presentation Start**

Open a media stream from the selected presentation source.

Requires user role: USER

**USAGE:**

```plaintext
xCommand Presentation Start [PresentationSource: PresentationSource] [SendingMode: SendingMode] [ConnectorId: ConnectorId] [Instance: Instance]
```

**PresentationSource:** Integer (2..3)

Select the video input source to be used for presentation, identified by source number.

**SendingMode:** LocalRemote/LocalOnly

Select whether the presentation is shown locally or locally and remotely.

**ConnectorId:** Integer (2..2)

Select the video input source to be used for presentation, identified by connectorId.

**Instance:** New/1/2/3/4/5/6

Select which local presentation instance you wish to start.

**xCommand Presentation Stop**

Stop the media stream from the presentation source.

Requires user role: USER

**USAGE:**

```plaintext
xCommand Presentation Stop [Instance: Instance] [PresentationSource: PresentationSource]
```

**Instance:** 1/2/3/4/5/6

Select which local presentation you wish to stop, identified by presentation instance.

**PresentationSource:** Integer (2..2)

Select which local presentation you wish to stop, identified by source number.
Provisioning commands

xCommand Provisioning CompleteUpgrade
Starts installing the software upgrade if you wish to install it before it is set to do so.
Requires user role: USER

USAGE:
  xCommand Provisioning CompleteUpgrade

xCommand Provisioning PostponeUpgrade
Postpones the installing of the software upgrade.
Requires user role: USER

USAGE:
  xCommand Provisioning PostponeUpgrade SecondsToPostpone
  where
  SecondsToPostpone: Integer (0..65534)
    Set how long to postpone the upgrade. The value is in seconds.

xCommand Provisioning CUCM CTL Delete
Delete the stored CTL and ITL files (CTL: Certificate Trust List, ITL: Identity Trust List).
Requires user role: USER

USAGE:
  xCommand Provisioning CUCM CTL Delete

xCommand Provisioning CUCM CTL Show
Shows the content of the installed Certificate Trust List file (CTL), if it exists. Each entry displayed contains the information about one specific certificate. If a certificate has been deleted in CTL, it is marked accordingly in the output.
Requires user role: USER

USAGE:
  xCommand Provisioning CUCM CTL Show

xCommand Provisioning StartUpgrade
The codec software can be upgraded from the provisioning server. When starting the upgrade the software is automatically downloaded and installed. The codec reboots to complete the software upgrade.
Requires user role: USER

USAGE:
  xCommand Provisioning StartUpgrade
Proximity commands

xCommand Proximity Services Activate
Reactivate the Proximity services that were deactivated with xCommand Proximity Services Deactivate.
Requires user role: USER

USAGE:
xCommand Proximity Services Activate

xCommand Proximity Services Deactivate
This command deactivates all proximity services on the endpoint. To reactivate proximity services use the command xCommand Proximity Services Activate.
Requires user role: USER

USAGE:
xCommand Proximity Services Deactivate

RoomPreset commands

xCommand RoomPreset Activate
Activate one of the locally stored presets.
Note that information about all video input sources, and pan, tilt, zoom and focus values for all cameras are included in the same preset. In contrast, the xCommand Camera Preset commands applies to individual cameras only.
Requires user role: USER

USAGE:
xCommand RoomPreset Activate PresetId: PresetId
where
PresetId: Integer (1..15)
The ID of the preset you want to activate.

xCommand RoomPreset Clear
Delete a preset.
Note that information about all video input sources, and pan, tilt, zoom and focus values for all cameras are included in the same preset. In contrast, the xCommand Camera Preset commands applies to individual cameras only.
Requires user role: USER

USAGE:
xCommand RoomPreset Clear PresetId: PresetId
where
PresetId: Integer (1..15)
The ID of the preset you want to delete.
xCommand RoomPreset Store

Store the connector selections for all video input sources and the current position (pan and tilt),
zoom and focus values for all cameras.

Note that information about all video input sources, and pan, tilt, zoom and focus values for
all cameras are included in the same preset. The system may hold 15 such predefined video
input presets. These presets are available for far end control, i.e. they are referred in the
PresetId parameter of the xCommand FarEndControl Preset Activate command. In contrast, the
xCommand Camera Preset commands applies to individual cameras only. Those presets are
not available for far end control.

Requires user role: USER

USAGE:

xCommand RoomPreset Store [Description: "Description"] PresetId: PresetId Type: Type

where

Description: String (0, 255)

Enter a description of the camera preset.

PresetId: Integer (1..15)

The ID of this preset.

Type: All/Camera

Not applicable. If you want to ensure that a preset only affects camera positions we
recommend that you select Camera.

Security commands

xCommand Security Persistency

Set the following features to persistent or non-persistent mode. In non-persistent mode
the information gathered by the specified feature does not persist a reboot of the system.
Persistent mode is the default. This command reboots the system.

Requires user role: ADMIN

USAGE:

xCommand Security Persistency Configurations: Configurations CallHistory: CallHistory InternalLogging: InternalLogging LocalPhonebook: LocalPhonebook DHCP: DHCP ConfirmAndReboot: ConfirmAndReboot

where

Configurations: NonPersistent/Persistent

In non-persistent mode, all configurations are set back to default when the system
reboots.

CallHistory: NonPersistent/Persistent

In non-persistent mode call history is deleted when the system reboots.

InternalLogging: NonPersistent/Persistent

In non-persistent mode eventlog is deleted when the system reboots.

LocalPhonebook: NonPersistent/Persistent

In non-persistent mode local phone book is deleted when the system reboots.

DHCP: NonPersistent/Persistent

In non-persistent mode all IP related information is deleted when the system reboots.

ConfirmAndReboot: Yes

Reboots the system.
Standby commands

**xCommand Standby Activate**
Set the system in standby mode, which turns off the video outputs and put the camera into sleep mode.
Requires user role: USER

**USAGE:**
```
xCommand Standby Activate
```

**xCommand Standby Deactivate**
Bring the system out of standby mode.
Requires user role: USER

**USAGE:**
```
xCommand Standby Deactivate
```

**xCommand Standby ResetTimer**
Set a temporary standby delay. If the system is in standby mode when the reset timer is set, the system is brought out of standby mode. When left idle for the given delay the system goes into standby mode. Setting the reset timer does not affect the Standby Delay in the Advanced configuration menu (or by xConfiguration Standby Delay). Next time this delay is the valid standby delay.
Requires user role: USER

**USAGE:**
```
xCommand Standby ResetTimer Delay: [Delay]
```

SystemUnit commands

**xCommand SystemUnit Boot**
Reboot the system.
Requires user role: USER

**USAGE:**
```
xCommand SystemUnit Boot [Action: Action]
```

**Action:** Restart/Shutdown
As a default the system restarts after a reboot. By selecting Shutdown, the system will not restart.

**xCommand SystemUnit FactoryReset**
Reset the codec to factory default settings. The call logs are deleted and all system parameters are reset to default values. All files that have been uploaded to the codec are deleted. Option key(s) are not affected.
As a default the system restarts after the factory reset, but other behaviour can be forced by selecting a different TrailingAction.
Requires user role: ADMIN

**USAGE:**
```
xCommand SystemUnit FactoryReset Confirm: Confirm [TrailingAction: TrailingAction]
```

**Confirm:** Yes
**TrailingAction:** NoAction/Restart/Shutdown
Select Shutdown or NoAction to override the default behaviour (Restart).
**xCommand SystemUnit Notifications RemoveAll**
Clears the list of system notifications that are reported by xStatus SystemUnit Notifications Text/Type.
Requires user role: ADMIN

**USAGE:**
```
xCommand SystemUnit Notifications RemoveAll
```

**xCommand SystemUnit OptionKey Add**
Add an option key to support additional features.
Requires user role: ADMIN

**USAGE:**
```
xCommand SystemUnit OptionKey Add Key: "Key"
```

**Key:** String (16, 24)
The key you have received for the option you wish to switch on.

**xCommand SystemUnit OptionKey Remove**
Remove a specified option key.
Requires user role: ADMIN

**USAGE:**
```
xCommand SystemUnit OptionKey Remove Type: Type
```

**Type:** Encryption/PremiumResolution/RemoteMonitoring

**xCommand SystemUnit OptionKey RemoveAll**
Remove all option keys.
Requires user role: ADMIN

**USAGE:**
```
xCommand SystemUnit OptionKey RemoveAll Confirm: Confirm
```

**Confirm:** Yes

**xCommand SystemUnit SoftwareUpgrade**
Initiate a software upgrade by fetching the software from a given URL. If the server requires username and password these parameters must be included.
Requires user role: ADMIN

**USAGE:**
```
xCommand SystemUnit SoftwareUpgrade URL: "URL" [UserName: "UserName"] [Password: "Password"]
```

**URL:** String (0, 255)
The software package location

**UserName:** String (0, 255)
User name to access the server location, if needed.

**Password:** String (0, 255)
Password to access the server location, if needed.
Time commands

**xCommand Time DateTime Get**
Read the time and date from the system.
Requires user role: USER

**USAGE:**
```
xCommand Time DateTime Get
```

**xCommand Time DateTime Set**
Set the date and time for the system, if not available from NTP (Network Time Protocol).
Requires user role: ADMIN

**USAGE:**
```
xCommand Time DateTime Set [Year: \_\_\_\_] [Month: \_\_\_\_] [Day: \_\_\_] [Hour: \_\_\_] [Minute: \_\_\_] [Second: \_\_\_]
```

where

**Year:** Integer (2015..2037)

**Month:** Integer (1..12)

**Day:** Integer (1..31)

**Hour:** Integer (0..23)

**Minute:** Integer (0..59)

**Second:** Integer (0..59)

UserInterface commands

**xCommand UserInterface Message Alert Clear**
Remove the message which was displayed using the xCommand Message Alert Display command. This is required when the Duration parameter is not set.
Requires user role: USER

**USAGE:**
```
xCommand UserInterface Message Alert Clear
```

**xCommand UserInterface Message Alert Display**
Display a message on screen, for a specified duration of time (in seconds).
NOTE: If Duration is not set, the command must be followed by xCommand Message Alert Clear.
Use the xFeedback commands to monitor the feedback from the user. Read more about the xFeedback commands in the API introduction section in this guide.
Requires user role: ADMIN

**USAGE:**
```
xCommand UserInterface Message Alert Display [Title: "Title"] Text: "Text" [Duration: \_\_\_\_]
```

where

**Title:** String (0, 255)
Enter a message title.

**Text:** String (0, 255)
Enter the message to be displayed. The <p> and <br> HTML tags will result in line breaks as normal; any other tags will appear as plain text.

**Duration:** Integer (0..3600)
Set how long (in seconds) the message is to be displayed on the screen. If set to 0 (zero) the message does not disappear until a xCommand Message Alert Clear message has been sent.
xCommand UserInterface Message Prompt Clear

Remove the window displayed using the xCommand Message Alert Display command. Use the xFeedback commands to monitor the feedback from the user. Read more about the xFeedback commands in the API introduction section in this guide.

Requires user role: USER

**USAGE:**

xCommand UserInterface Message Prompt Clear [FeedbackId: "FeedbackId"]

where

**FeedbackId**: String (0, 255)

The FeedbackId corresponds to the FeedbackId given by the xCommand Message Prompt Display command.

---

xCommand UserInterface Message Prompt Display

Display a small window on screen with a title, text and up to five options for response from the user. The message is displayed on screen until the user gives a response, or until the system receives the following command xCommand Message Prompt Clear. Use the xFeedback commands to monitor the feedback from the user. Read more about the xFeedback commands in the API introduction section in this guide.

Requires user role: ADMIN

**USAGE:**

xCommand UserInterface Message Prompt Display [Title: "Title"] Text: "Text" [FeedbackId: "FeedbackId"] [Option.1: "Option.1"] [Option.2: "Option.2"] [Option.3: "Option.3"] [Option.4: "Option.4"] [Option.5: "Option.5"]

where

**Title**: String (0, 255)

Enter the message title.

**Text**: String (0, 255)

Enter the text line to be displayed. The <p> and <br> HTML tags will result in line breaks as normal; any other tags will appear as plain text.

**FeedbackId**: String (0, 255)

To identify the feedback enter a FeedbackId.

**Option.1**: String (0, 255)

Enter the text to appear on the feedback options.

**Option.2**: String (0, 255)

Enter the text to appear on the feedback options.

**Option.3**: String (0, 255)

Enter the text to appear on the feedback options.

**Option.4**: String (0, 255)

Enter the text to appear on the feedback options.
Option.5: String (0, 255)

Enter the text to appear on the feedback options.

xCommand UserInterface Message Prompt Response

Give a response to the xCommand Message Prompt Display.

Use the xFeedback commands to monitor the feedback from the user. Read more about the xFeedback commands in the API introduction section in this guide.

Requires user role: USER

USAGE:

xCommand UserInterface Message Prompt Response [FeedbackId: "FeedbackId"]

OptionId: OptionId

where

FeedbackId: String (0, 255)

The FeedbackId corresponds to the FeedbackId given by the xCommand Message Prompt Display command.

OptionId: Integer (1..5)

The OptionId corresponds to the OptionIds given as possible responses in the xCommand Message Prompt Display command.

xCommand UserInterface Message TextLine Clear

Clears the text line which was defined by the xCommand Message TextLine Display command.

Requires user role: USER

USAGE:

xCommand UserInterface Message TextLine Clear

xCommand UserInterface Message TextLine Display

Display a text line on screen. Optionally you can place the text line at a specified location and for a specified duration of time (in seconds).

NOTE: If Duration is not set, the command must be followed by xCommand Message TextLine Clear.

Requires user role: ADMIN

USAGE:

xCommand UserInterface Message TextLine Display Text: "Text" [X: X] [Y: Y] [Duration: Duration]

where

Text: String (0, 140)

Enter the text line to be displayed. The <p> and <br> HTML tags will result in line breaks as normal; any other tags will appear as plain text.

X: Integer (1..10000)

Enter the X-coordinate (horizontal) on screen. X=0 is in the upper left corner.

Y: Integer (1..10000)

Enter the Y-coordinate (vertical) on screen. Y=0 is in the upper left corner.

Duration: Integer (0..3600)

Set how long (in seconds) the text line is to be displayed on the screen. If set to 0 (zero) the text line is displayed until a xCommand Message TextLine Clear command has been sent.

xCommand UserInterface OSD Key Click

Emulates a remote control key press, followed by a key release.

Requires user role: ADMIN

USAGE:

xCommand UserInterface OSD Key Click Key: Key

where

Key: 0/1/2/3/4/5/6/7/8/9/C/Call/Disconnect/Down/F1/F2/F3/F4/F5/Grab/Home/Layout/Left/Mute/MuteMic/Ok/PhoneBook/Presentation/Right/Selfview/Square/SrcAux/SrcCamera/SrcDocCam/SrcPc/SrcVcr/Star/Up/VolumeDown/VolumeUp/ZoomIn/ZoomOut

Define the remote key to press.
xCommand UserInterface OSD Key Press
Emulates a remote control key press without releasing it. The Key Press command must be followed by a Key Release command to emulate releasing the key.
Requirements user role: ADMIN

**USAGE:**
```plaintext
xCommand UserInterface OSD Key Press Key: Key
```
where
```
Key: 0/1/2/3/4/5/6/7/8/9/C/Call/Disconnect/Down/F1/F2/F3/F4/F5/Grab/Home/Layout/Left/
Mute/MuteMic/Ok/PhoneBook/Presentation/Right/Selfview/Square/SrcAux/SrcCamera/
SrcDocCam/SrcPc/SrcVcr/Star/Up/VolumeDown/VolumeUp/ZoomIn/ZoomOut
```
Define the remote key to press.

xCommand UserInterface OSD Key Release
Emulates a remote control key release. The Key Release command is issued after a Key Press command.
Requirements user role: ADMIN

**USAGE:**
```plaintext
xCommand UserInterface OSD Key Release Key: Key
```
where
```
Key: 0/1/2/3/4/5/6/7/8/9/C/Call/Disconnect/Down/F1/F2/F3/F4/F5/Grab/Home/Layout/Left/
Mute/MuteMic/Ok/PhoneBook/Presentation/Right/Selfview/Square/SrcAux/SrcCamera/
SrcDocCam/SrcPc/SrcVcr/Star/Up/VolumeDown/VolumeUp/ZoomIn/ZoomOut
```
Define the remote control key to release.

UserManagement commands

xCommand UserManagement RemoteSupportUser Create
Create a remote support user passphrase that Technical Assistance Center (TAC) can use to access the system for troubleshooting.
Requirements user role: ADMIN

**USAGE:**
```plaintext
```
where
```
ExpiryDays: Integer (1..31)
```
Define the duration for the passphrase validity. Default is 7 days.

xCommand UserManagement RemoteSupportUser Delete
Delete the remote support user created with the command xCommand UserManagement RemoteSupportUser Create.
Requirements user role: ADMIN

**USAGE:**
```plaintext
xCommand UserManagement RemoteSupportUser Delete
```

xCommand UserManagement RemoteSupportUser DisablePermanently
Disable the creation of new remote support users. To enable the remote support user again you must factory reset your system.
Requirements user role: ADMIN

**USAGE:**
```plaintext
xCommand UserManagement RemoteSupportUser DisablePermanently Confirm: Confirm
```
where
```
Confirm: Yes
```
xCommand UserManagement RemoteSupportUser GetState
Retrieves the state of the generated remote support user, if one exists.
Requires user role: ADMIN

USAGE:
  xCommand UserManagement RemoteSupportUser GetState

xCommand UserManagement User Passphrase Change
Change the passphrase for the user you logged in as. If you are logged in as the administrator, this will change the administrator passphrase.
Requires user role: USER

USAGE:
  xCommand UserManagement User Passphrase Change NewPassphrase: "NewPassphrase"
  OldPassphrase: "OldPassphrase"
  Username: "Username"
  YourPassphrase: "YourPassphrase"

where

NewPassphrase: String (0, 255)

OldPassphrase: String (0, 255)

Username: String (0, 127)

YourPassphrase: String (0, 255)

xCommand UserManagement User Passphrase Set
Set a user passphrase for the specified user. You must be logged in as an administrator to set a user passphrase.
Requires user role: ADMIN

USAGE:
  xCommand UserManagement User Passphrase Set NewPassphrase: "NewPassphrase"
  Username: "Username"
  YourPassphrase: "YourPassphrase"

where

NewPassphrase: String (0, 255)

Username: String (0, 127)

YourPassphrase: String (0, 255)
Video commands

**xCommand Video ActiveSpeakerPIP Set**
Sets position for the active speakers PiP (picture in picture).
Requires user role: USER

**USAGE:**

```
xCommand Video ActiveSpeakerPIP Set Position: Position
```

where

```
Position: CenterLeft/CenterRight/LowerLeft/LowerRight/UpperCenter/UpperLeft/UpperRight
```
Select one of the predefined positions.

**xCommand Video Input Source SetActiveConnector**
Select which connector is active.
Requires user role: USER

**USAGE:**

```
xCommand Video Input Source SetActiveConnector [ConnectorId: ConnectorId]
```

where

```
ConnectorId: Integer (2..3)
```

The identifier (ID) of the connector. Connector [n] has ID n. Run the following API command to find the ID: xStatus Video Input Connector. The connector ID is also printed on the codec connector panel.

**xCommand Video Layout LayoutFamily Set**
Select the screen layout mode.
Requires user role: USER

**USAGE:**

```
xCommand Video Layout LayoutFamily Set [Target: Target] [: ] LayoutFamily: LayoutFamily [CustomLayoutName: "CustomLayoutName"]
```

where

```
Target: local/remote
```
Select if the target is the local layout or the remote layout.

```
: Integer (0..65534)
```
The CallID is returned when the xCommand Dial command is run. During the call you can run the xStatus Call command to see the CallId.

```
LayoutFamily: auto/custom/equal/overlay/prominent/single
```
Select a layout family.

```
CustomLayoutName: String (1, 128)
```
Enter a name for the layout.

**xCommand Video PresentationPIP Set**
Sets position for the presentation PiP (picture in picture).
Requires user role: USER

**USAGE:**

```
xCommand Video PresentationPIP Set Position: Position
```

where

```
Position: CenterLeft/CenterRight/LowerLeft/LowerRight/UpperCenter/UpperLeft/UpperRight
```
Select one of the predefined positions.
xCommand Video Selfview Set

Sets self-view on/off and specifies its size and position. If the parameter is not specified, current value is used.

Requires user role: USER

USAGE:

xCommand Video Selfview Set [Mode: Mode] [FullscreenMode: FullscreenMode] [PIPPosition: PIPPosition] [OnMonitorRole: OnMonitorRole]

where

Mode: On/Off
Selfview is set to on or off.

FullscreenMode: On/Off
Choose between displaying the self-view in full screen or as picture-in-picture.

PIPPosition: CenterLeft/CenterRight/LowerLeft/LowerRight/UpperCenter/UpperLeft/UpperRight
Select the position for the self-view image.

OnMonitorRole: First/Fourth/Second/Third
Displays self-view on monitors with this role.
Chapter 5

xStatus commands
**Description of the xStatus commands**

In this chapter, you can find all of the xStatus commands and the responses. Status type commands return information about the system and system processes. You can query all information or just some of it.

We recommend you visit our web site regularly for updated versions of the manual.

Go to: [http://www.cisco.com/go/sx-docs](http://www.cisco.com/go/sx-docs)

<table>
<thead>
<tr>
<th>Audio status</th>
<th>114</th>
</tr>
</thead>
<tbody>
<tr>
<td>xStatus Audio</td>
<td>114</td>
</tr>
<tr>
<td>xStatus Audio Input Connectors HDMI</td>
<td>114</td>
</tr>
<tr>
<td>xStatus Audio Connectors Microphone</td>
<td>114</td>
</tr>
<tr>
<td>xStatus Audio Input Connectors Microphone</td>
<td>114</td>
</tr>
<tr>
<td>xStatus Audio Input Remotelnput [n] CallId</td>
<td>114</td>
</tr>
<tr>
<td>xStatus Audio Microphones Mute</td>
<td>114</td>
</tr>
<tr>
<td>xStatus Audio Volume</td>
<td>115</td>
</tr>
<tr>
<td>xStatus Audio VolumeMute</td>
<td>115</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Call status</th>
<th>115</th>
</tr>
</thead>
<tbody>
<tr>
<td>xStatus Call [n]</td>
<td>115</td>
</tr>
<tr>
<td>xStatus Call [n] AnswerState</td>
<td>115</td>
</tr>
<tr>
<td>xStatus Call [n] AttendedTransferFrom</td>
<td>115</td>
</tr>
<tr>
<td>xStatus Call [n] CallbackNumber</td>
<td>115</td>
</tr>
<tr>
<td>xStatus Call [n] CallType</td>
<td>116</td>
</tr>
<tr>
<td>xStatus Call [n] DeviceType</td>
<td>116</td>
</tr>
<tr>
<td>xStatus Call [n] Direction</td>
<td>116</td>
</tr>
<tr>
<td>xStatus Call [n] DisplayName</td>
<td>116</td>
</tr>
<tr>
<td>xStatus Call [n] Duration</td>
<td>116</td>
</tr>
<tr>
<td>xStatus Call [n] Encryption Type</td>
<td>116</td>
</tr>
<tr>
<td>xStatus Call [n] FacilityServiced</td>
<td>117</td>
</tr>
<tr>
<td>xStatus Call [n] HoldReason</td>
<td>117</td>
</tr>
<tr>
<td>xStatus Call [n] PlacedOnHold</td>
<td>117</td>
</tr>
<tr>
<td>xStatus Call [n] Protocol</td>
<td>117</td>
</tr>
<tr>
<td>xStatus Call [n] ReceiveCallRate</td>
<td>117</td>
</tr>
<tr>
<td>xStatus Call [n] RemoteNumber</td>
<td>117</td>
</tr>
<tr>
<td>xStatus Call [n] Status</td>
<td>118</td>
</tr>
<tr>
<td>xStatus Call [n] TransmitCallRate</td>
<td>118</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cameras status</th>
<th>118</th>
</tr>
</thead>
<tbody>
<tr>
<td>xStatus Cameras Camera</td>
<td>118</td>
</tr>
<tr>
<td>xStatus Cameras Camera [n] Capabilities Options</td>
<td>118</td>
</tr>
<tr>
<td>xStatus Cameras Camera [n] Connected</td>
<td>118</td>
</tr>
<tr>
<td>xStatus Cameras Camera [n] Flip</td>
<td>118</td>
</tr>
<tr>
<td>xStatus Cameras Camera [n] MacAddress</td>
<td>119</td>
</tr>
<tr>
<td>xStatus Cameras Camera [n] Manufacturer</td>
<td>119</td>
</tr>
<tr>
<td>xStatus Cameras Camera [n] Model</td>
<td>119</td>
</tr>
<tr>
<td>xStatus Cameras Camera [n] SerialNumber</td>
<td>119</td>
</tr>
<tr>
<td>xStatus Cameras Camera [n] SoftwareID</td>
<td>119</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Capabilities status</th>
<th>120</th>
</tr>
</thead>
<tbody>
<tr>
<td>xStatus Capabilities</td>
<td>120</td>
</tr>
<tr>
<td>xStatus Capabilities Conference MaxActiveCalls</td>
<td>120</td>
</tr>
<tr>
<td>xStatus Capabilities Conference MaxAudioCalls</td>
<td>120</td>
</tr>
<tr>
<td>xStatus Capabilities Conference MaxCalls</td>
<td>120</td>
</tr>
<tr>
<td>xStatus Capabilities Conference MaxVideoCalls</td>
<td>120</td>
</tr>
</tbody>
</table>
### Conference status
- xStatus Conference
- xStatus Conference ActiveSpeaker CallId
- xStatus Conference Call [n] BookingId
- xStatus Conference Call [n] Capabilities FarendMessage Mode
- xStatus Conference Call [n] Capabilities FECC Mode
- xStatus Conference Call [n] Capabilities FECC NumberOfPresets
- xStatus Conference Call [n] Capabilities FECC NumberOfSources
- xStatus Conference Call [n] Capabilities FECC Source [n] Name
- xStatus Conference Call [n] Capabilities FECC Source [n] Options
- xStatus Conference Call [n] Capabilities FECC Source [n] SourceId
- xStatus Conference Call [n] Capabilities Hold
- xStatus Conference Call [n] Capabilities lxChannel Status
- xStatus Conference Call [n] Capabilities Multipoint Mode
- xStatus Conference Presentation CallId
- xStatus Conference Presentation LocalInstance [n] SendingMode
- xStatus Conference Presentation LocalInstance [n] Source
- xStatus Conference Presentation Mode
- xStatus Conference SpeakerLock CallId
- xStatus Conference SpeakerLock Mode

### Diagnostics status
- xStatus Diagnostics
- xStatus Diagnostics Message [n] Description
- xStatus Diagnostics Message [n] Level
- xStatus Diagnostics Message [n] References
- xStatus Diagnostics Message [n] Type

### H323 status
- xStatus H323
- xStatus H323 Gatekeeper Address
- xStatus H323 Gatekeeper Port
- xStatus H323 Gatekeeper Reason
- xStatus H323 Gatekeeper Status
- xStatus H323 Mode Reason
- xStatus H323 Mode Status

### HttpFeedback status
- xStatus HttpFeedback
- xStatus HttpFeedback [1..4] Expression [1..15]
- xStatus HttpFeedback [1..4] URL

### Network status
- xStatus Network
- xStatus Network 1 CDP Address
- xStatus Network 1 CDP Capabilities
- xStatus Network 1 CDP DeviceId
- xStatus Network 1 CDP Duplex
- xStatus Network 1 CDP Platform
- xStatus Network 1 CDP PortId
- xStatus Network 1 CDP PrimaryMgmtAddress
- xStatus Network 1 CDP SysName
- xStatus Network 1 CDP SysObjectID
- xStatus Network 1 CDP Version
- xStatus Network 1 CDP VTPMgmtDomain
- xStatus Network 1 DNS Domain Name
- xStatus Network 1 DNS Server [1..5] Address
- xStatus Network 1 Ethernet MacAddress
- xStatus Network 1 Ethernet Speed
- xStatus Network 1 IPv4 Address
- xStatus Network 1 IPv4 Gateway
- xStatus Network 1 IPv4 SubnetMask
<table>
<thead>
<tr>
<th>Status</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Services</td>
<td>132</td>
</tr>
<tr>
<td>Peripherals</td>
<td>133</td>
</tr>
<tr>
<td>Provisioning</td>
<td>134</td>
</tr>
<tr>
<td>Security</td>
<td>137</td>
</tr>
<tr>
<td>SIP</td>
<td>138</td>
</tr>
</tbody>
</table>

### Network Services
- Status Network Services NTP Current Address
- Status NetworkServices NTP Server [n] Address
- Status NetworkServices NTP Status

### Peripherals
- Status Peripherals ConnectedDevice [n] HardwareInfo
- Status Peripherals ConnectedDevice [n] ID
- Status Peripherals ConnectedDevice [n] Name
- Status Peripherals ConnectedDevice [n] SoftwareInfo
- Status Peripherals ConnectedDevice [n] Status
- Status Peripherals ConnectedDevice [n] Type

### Provisioning
- Status Provisioning Software Current CompletedAt
- Status Provisioning Software Current URL
- Status Provisioning Software Current VersionId
- Status Provisioning Software UpgradeStatus LastChange
- Status Provisioning Software UpgradeStatus Message
- Status Provisioning Software UpgradeStatus Phase
- Status Provisioning Software UpgradeStatus SessionId
- Status Provisioning Software UpgradeStatus Status
- Status Provisioning Software UpgradeStatus URL
- Status Provisioning Software UpgradeStatus VersionId

### Security
- Status Security Persistency CallHistory
- Status Security Persistency Configurations
- Status Security Persistency DHCP
- Status Security Persistency InternalLogging
- Status Security Persistency LocalPhonebook

### SIP
- Status SIP AlternateURI Alias [n] URI
- Status SIP AlternateURI Primary [n] URI
- Status SIP Authentication
- Status SIP CallForward DisplayName
- Status SIP CallForward Mode
- Status SIP CallForward URI
- Status SIP Mailbox MessagesWaiting
- Status SIP Mailbox URI
- Status SIP Proxy [n] Address
- Status SIP Proxy [n] Secure
- Status SIP Proxy [n] Status
- Status SIP Proxy [n] Verified
- Status SIP Registration [n] Authentication
- Status SIP Registration [n] Reason

---

xStatus Provisioning Status

---

Proximity status

---

xStatus Provisioning Status

---

Security status

---

xStatus Provisioning Status

---

SIP status

---

xStatus Provisioning Status

---
Standby status

xStatus Standby

xStatus Standby State

SystemUnit status

xStatus SystemUnit

xStatus SystemUnit Hardware Module SerialNumber

xStatus SystemUnit Hardware Monitoring Fan [n] Status

xStatus SystemUnit Hardware Temperature

xStatus SystemUnit Notifications Notification [n] Text

xStatus SystemUnit Notifications Notification [n] Type

xStatus SystemUnit ProductId

xStatus SystemUnit ProductId

xStatus SystemUnit ProductId

xStatus SystemUnit ProductId

xStatus SystemUnit ProductId

xStatus SystemUnit ProductId

xStatus SystemUnit Software Name

xStatus SystemUnit Software OptionKeys Encryption

xStatus SystemUnit Software OptionKeys PremiumResolution

xStatus SystemUnit Software OptionKeys RemoteMonitoring

xStatus SystemUnit Software ReleaseDate

xStatus SystemUnit Software Version

xStatus SystemUnit State NumberOfActiveCalls

xStatus SystemUnit State NumberOfInProgressCalls

xStatus SystemUnit State NumberOfSuspendedCalls

xStatus SystemUnit Uptime

Time status

xStatus Time

xStatus Time SystemTime

UserInterface status

xStatus UserInterface

xStatus UserInterface ContactInfo ContactMethod [n] Number

xStatus UserInterface ContactInfo Name

Video status

xStatus Video

xStatus Video ActiveSpeaker PIPPosition

xStatus Video Input

xStatus Video Input Connector [n] Connected

xStatus Video Input Connector [n] SignalState

xStatus Video Input Connector [n] SourceId

xStatus Video Input Connector [n] Type

xStatus Video Input MainVideoSource

xStatus Video Input Source [n] ConnectorId

xStatus Video Input Source [n] FormatStatus

xStatus Video Input Source [n] FormatType

xStatus Video Input Source [n] MediaChannelId

xStatus Video Input Source [n] Resolution Height

xStatus Video Input Source [n] Resolution RefreshRate

xStatus Video Input Source [n] Resolution Width

xStatus Video Monitors

xStatus Video Output

xStatus Video Output Connector [n] Connected

xStatus Video Output Connector [n] ConnectedDevice CEC DeviceType

xStatus Video Output Connector [n] ConnectedDevice CEC DeviceType

xStatus Video Output Connector [n] ConnectedDevice CEC PowerStatus

xStatus Video Output Connector [n] ConnectedDevice Name

xStatus Video Output Connector [n] ConnectedDevice PreferredFormat

xStatus Video Output Connector [n] MonitorRole

xStatus Video Output Connector [n] Resolution Height

xStatus Video Output Connector [n] Resolution RefreshRate

xStatus Video Output Connector [n] Resolution Width

xStatus Video Output Connector [n] Type

xStatus Video Presentation PIPPosition

xStatus Video Selfview FullscreenMode

xStatus Video Selfview Mode

xStatus Video Selfview OnMonitorRole

xStatus Video Selfview PIPPosition
Audio status

xStatus Audio
Shows the top level overview of the audio status. The identities of the LocalInput, RemoteInput, LocalOutput and RemoteOutput are used when querying additional information.

xStatus Audio Input Connectors HDMI [n] EcReferenceDelay
Returns the detected latency for each loudspeaker to microphone path for systems supporting HDMI input used as microphone input. The result is in milliseconds.

Value space of the result returned:
Integer

Example:
xStatus Audio Input Connectors HDMI 1 EcReferenceDelay
*s Audio Connectors HDMI 1 EcReferenceDelay: 0
** end

xStatus Audio Input Connectors Microphone [n] ConnectionStatus
Indicates whether a microphone is detected on the microphone input connector.

Value space of the result returned:
Connected/NotConnected/Unknown

Example:
xStatus Audio Input Connectors Microphone ConnectionStatus
*s Audio Input Connectors Microphone 1 ConnectionStatus: NotConnected
*s Audio Input Connectors Microphone 2 ConnectionStatus: Connected
** end

xStatus Audio Input RemoteInput [n] CallId
Shows the CallId for the remote audio input.
You can run the command xStatus Audio Input RemoteInput to find the identity [n] of the input.

Value space of the result returned:
0..65534

Example:
xStatus Audio Input RemoteInput CallId
*s Audio Input RemoteInput 8 CallId: 28
** end

xStatus Audio Microphones Mute
Shows whether the microphones are muted.

Value space of the result returned:
On/Off

Example:
xStatus Audio Microphones Mute
*s Audio Microphones Mute: Off
** end
xStatus Audio Volume
Shows the volume level (dB) of the loudspeaker output.

Value space of the result returned:
0..100

Example:
xStatus Audio Volume
*s Audio Volume: 70
** end

xStatus Audio VolumeMute
Shows whether the endpoint volume is set to mute.

Value space of the result returned:
Off/On

Example:
xStatus Audio VolumeMute
*s Audio VolumeMute: Off
** end

Call status

xStatus Call [n]
Shows the top level overview of the call status. The call identity is used when query for additional information about the call.

xStatus Call [n] AnswerState
Indicates if a call is answered, ignored or has been automatically answered by a system.

Value space of the result returned:
Unanswered/Ignored/Autoanswered/Answered

Example:
xStatus Call AnswerState
*s Call 5 AnswerState: Answered
** end

xStatus Call [n] AttendedTransferFrom
Shows the CallId for the call the current call was transferred from.

Value space of the result returned:
Integer

Example:
xStatus Call 1 AttendedTransferFrom
*s Call 1 AttendedTransferFrom: 1234
** end

xStatus Call [n] CallbackNumber
Shows the remote (far end) number or URI of an incoming or outgoing call, including the call protocol, for call back. You can run the command xStatus Call to find the call identity.

Value space of the result returned:
String

Example:
xStatus Call 27 CallbackNumber
*s Call 27 CallbackNumber: "h323:firstname.lastname@company.com"
** end
xStatus Call [n] CallType
Shows the call type of the incoming or outgoing call. You can run the command xStatus Call to find the call identity.

Value space of the result returned:
Video/Audio/AudioCanEscalate/ForwardAllCall/Unknown

Example:
xStatus Call 27 CallType
*s Call 27 CallType: Video
** end

xStatus Call [n] DeviceType
Indicates whether the remote system is a single endpoint or an MCU. Some Cisco endpoints (SX20, SX80, MX200 G2, MX300 G2) have built-in MCU capabilities.

Value space of the result returned:
Endpoint/MCU

Example:
xStatus Call DeviceType
*s Call 4 DeviceType: Endpoint
** end

xStatus Call [n] Direction
States the direction of the call initiation. You can run the command xStatus Call to find the call identity.

Value space of the result returned:
Incoming/Outgoing

Example:
xStatus Call 27 Direction
*s Call 27 Direction: Outgoing
** end

xStatus Call [n] DisplayName
Shows the name of the remote (far end) participant in an incoming or outgoing call. You can run the command xStatus Call to find the call identity.

Value space of the result returned:
String

Example:
xStatus Call 27 DisplayName
*s Call 27 DisplayName: "firstname.lastname@company.com"
** end

xStatus Call [n] Duration
Shows the duration of a call (in seconds). You can run the command xStatus Call to find the call identity.

Value space of the result returned:
Integer

Example:
xStatus Call 27 Duration
*s Call 27 Duration: 2354
** end

xStatus Call [n] Encryption Type
Shows the encryption type of the call. You can run the command xStatus Call to find the call identity.

Value space of the result returned:
None/Aes-128

Example:
xStatus Call 27 Encryption Type
*s Call 27 Encryption Type: "None"
** end
xStatus Call [n] FacilityServiceId
When calling a facility service, the facility service id is shown. Otherwise the value 0 is returned.

Value space of the result returned:
0..5

Example:
```
xStatus Call FacilityServiceId
*\s Call 3 FacilityServiceId: 1
** end
```

xStatus Call [n] HoldReason
Shows the reason the current outgoing call was put on hold.
Conference: On hold while the call is being merged into a conference.
Transfer: On hold while the call is being transferred.
None: All other instances.

Value space of the result returned:
Conference/Transfer/None

Example:
```
xStatus Call HoldReason
*\s Call 2 HoldReason: None
** end
```

xStatus Call [n] PlacedOnHold
Shows the placed on hold status of the call. You can run the command xStatus Call to find the call identity.

Value space of the result returned:
True/False

Example:
```
xStatus Call PlacedOnHold
*\s Call 27 PlacedOnHold: False
** end
```

xStatus Call [n] Protocol
Shows the call protocol of the incoming or outgoing call. You can run the command xStatus Call to find the call identity.

Value space of the result returned:
H320/H323/SIP

Example:
```
xStatus Call Protocol
*\s Call 27 Protocol: "h323"
** end
```

xStatus Call [n] ReceiveCallRate
Shows the receive bandwidth in the call in kilobits per second (kbps). You can run the command xStatus Call to find the call identity.

Value space of the result returned:
Integer

Example:
```
xStatus Call ReceiveCallRate
*\s Call 27 ReceiveCallRate: 4000
** end
```

xStatus Call [n] RemoteNumber
Shows the remote (far end) number or URI of an incoming or outgoing call. You can run the command xStatus Call to find the call identity.

Value space of the result returned:
String

Example:
```
xStatus Call RemoteNumber
*\s Call 27 RemoteNumber: "5585232"
** end
```
xStatus Call [n] Status
Shows the status of a call. You can run the command xStatus Call to find the call identity.

Value space of the result returned:
Idle/Dialling/Ringing/Connecting/Connected/Disconnecting/OnHold/EarlyMedia/Preserved/
RemotePreserved

Example:
xStatus Call 27 Status
*s Call 27 Status: Connected
** end

xStatus Call [n] TransmitCallRate
Shows the transmit bandwidth in the call in kilobits per second (kbps). You can run the command xStatus Call to find the call identity.

Value space of the result returned:
Integer

Example:
xStatus Call 27 TransmitCallRate
*s Call 27 TransmitCallRate: 768
** end

Cameras status

xStatus Cameras Camera
Shows the top level overview of the camera status.

xStatus Cameras Camera [n] Capabilities Options
Shows the camera capabilities (ptzf = pan, tilt, zoom, focus).

Value space of the result returned:
String

Example:
xStatus Cameras Camera 1 Capabilities Options
*s Camera 1 Capabilities Options: "ptzf"
** end

xStatus Cameras Camera [n] Connected
Shows if the camera is connected or not.

Value space of the result returned:
True/False

Example:
xStatus Cameras Camera 1 Connected
*s Camera 1 Connected: True
** end

xStatus Cameras Camera [n] Flip
In Flip mode (vertical flip) the image can be flipped upside down.

Value space of the result returned:
Auto/On/Off

Example:
xStatus Cameras Camera 1 Flip
*s Camera 1 Flip: "Off"
** end
xStatus Cameras Camera [n] MacAddress
Shows the MAC (Media Access Control) address for the camera.

Value space of the result returned:
String

Example:
xStatus Cameras Camera 1 MacAddress
*s Camera 1 MacAddress: ""
** end

xStatus Cameras Camera [n] Manufacturer
Shows the manufacturer of the camera.

Value space of the result returned:
String

Example:
xStatus Cameras Camera 1 Manufacturer
*s Camera 1 Manufacturer: "Cisco"
** end

xStatus Cameras Camera [n] Model
Shows the camera model.

Value space of the result returned:
String

Example:
xStatus Cameras Camera 1 Model
*s Camera 1 Model: "Precision 40"
** end

xStatus Cameras Camera [n] SerialNumber
Shows the camera serial number.

Value space of the result returned:
String

Example:
xStatus Cameras Camera 1 SerialNumber
*s Camera 1 SerialNumber: "B1AB26B00010"
** end

xStatus Cameras Camera [n] SoftwareID
Shows the software identity of the camera.

Value space of the result returned:
String

Example:
xStatus Cameras Camera 1 SoftwareID
*s Camera 1 SoftwareID: "S01718-4.0FINAL [ID:40063] 2014-10-20"
** end
Capabilities status

xStatus Capabilities
Shows the top level overview of the capabilities status.

xStatus Capabilities Conference MaxActiveCalls
Shows the maximum number of simultaneous active calls. Calls that are set on hold/transfer are not counted as active.

Value space of the result returned:
0..5

Example:
```
xStatus Capabilities Conference MaxNumberOfActiveCalls
's Capabilities Conference MaxNumberOfActiveCalls: 3
** end
```

xStatus Capabilities Conference MaxAudioCalls
Shows the maximum number of simultaneous audio calls that is supported.

Value space of the result returned:
Integer

Example:
```
xStatus Capabilities Conference MaxAudioCalls
's Capabilities Conference MaxAudioCalls: 3
** end
```

xStatus Capabilities Conference MaxCalls
Shows the maximum number of simultaneous calls.

Value space of the result returned:
0..5

Example:
```
xStatus Capabilities Conference MaxCalls
's Capabilities Conference MaxCalls: 3
** end
```
Conference status

xStatus Conference
Shows the top level overview of the conference status. The identity of the Conference Call can only be read during a call.

xStatus Conference ActiveSpeaker CallId
Shows the CallId of the current active speaker.
Value space of the result returned:
Integer
Example:
  xStatus Conference ActiveSpeaker CallId
  *s Conference ActiveSpeaker CallId: 3
  ** end

xStatus Conference Call [n] BookingId
Shows the booking ID of a conference (if assigned). The booking ID can be used for easy identification of a call or conference.
Value space of the result returned:
String
Example:
  xStatus Conference Call 2 BookingId
  *s Conference Call 2 BookingId: "MyConference"
  ** end

xStatus Conference Call [n] Capabilities FarendMessage Mode
Shows whether or not you have permission to control the input sources at a far end site.
On: Far end input source control is permitted.
Off: Far end input source control is not permitted.
Value space of the result returned:
On/Off
Example:
  xStatus Conference Call 2 Capabilities FarendMessage Mode
  *s Conference Call 2 Capabilities FarendMessage Mode: On
  ** end

xStatus Conference Call [n] Capabilities FECC Mode
Not applicable in this release.
Value space of the result returned:
On/Off
Example:
  xStatus Conference Call 2 Capabilities FECC Mode
  *s Conference Call 2 Capabilities FECC Mode: Off
  ** end

xStatus Conference Call [n] Capabilities FECC NumberOfPresets
Shows the number of presets available for the input sources at a far end site.
Value space of the result returned:
1..15
Example:
  xStatus Conference Call 2 Capabilities FECC NumberOfPresets
  *s Conference Call 2 Capabilities FECC NumberOfPresets: 15
  ** end

xStatus Conference Call [n] Capabilities FECC NumberOfSources
Shows the number of input sources that can be connected at a far end site.
Value space of the result returned:
1..5
Example:
  xStatus Conference Call 2 Capabilities FECC NumberOfSources
  *s Conference Call 2 Capabilities FECC NumberOfSources: 5
  ** end
xStatus Conference Call [n] Capabilities FECC Source [n] Name

Shows the name of an input source that can be connected at a far end site.

Value space of the result returned:
String

Example:
```
xStatus Conference Call 2 Capabilities FECC Source 1 Name
's Conference Call 2 Capabilities FECC Source 1 Name: "Main camera"
** end
```

xStatus Conference Call [n] Capabilities FECC Source [n] Options

Shows available options for an input source that can be connected at a far end site (for a camera: p=pan; t=tilt; z=zoom; f=focus).

Value space of the result returned:
String

Example:
```
xStatus Conference Call 2 Capabilities FECC Source 1 Options
's Conference Call 2 Capabilities FECC Source 1 Options: "ptzf"
** end
```

xStatus Conference Call [n] Capabilities FECC Source [n] SourceId

Shows the ID of an input source that can be connected at a far end site.

Value space of the result returned:
Integer

Example:
```
xStatus Conference Call 2 Capabilities FECC Source 1 SourceId
's Conference Call 2 Capabilities FECC Source 1 SourceId: 6
** end
```

xStatus Conference Call [n] Capabilities Hold

Indicates whether the far-end site can be placed on hold or not.

Value space of the result returned:
True/False

Example:
```
xStatus Conference Call Capabilities Hold
's Conference Call 2 Capabilities Hold: True
** end
```

xStatus Conference Call [n] Capabilities IxChannel Status

Not applicable in this release.

Value space of the result returned:
Active/Failed/Off

Example:
```
xStatus Conference Call 4 Capabilities IxChannel Status
's Conference Call 4 Capabilities IxChannel Status: Active
** end
```

xStatus Conference Call [n] Capabilities Presentation

Lists the presentation capabilities for other participants in the conference.

Value space of the result returned:
True/False

Example:
```
xStatus Conference Call 2 Capabilities Presentation
's Conference Call 2 Capabilities Presentation: True
** end
```
xStatus Conference Call [n] Manufacturer
Shows the manufacturer of the video system at a far end site.

Value space of the result returned:
String

Example:
xStatus Conference Call 2 Manufacturer
's Conference Call 2 Manufacturer: "Cisco"
** end

xStatus Conference Call [n] MicrophonesMuted
Lists the audio mute status for other participants in the conference.

Value space of the result returned:
True/False

Example:
xStatus Conference Call 2 MicrophonesMuted
's Conference Call 2 MicrophonesMuted: True
** end

xStatus Conference Call [n] SoftwareID
Shows the ID of the software running of the video system at a far end site.

Value space of the result returned:
String

Example:
xStatus Conference Call 2 SoftwareID
's Conference Call 2 SoftwareID: "CE8"
** end

xStatus Conference DoNotDisturb
Shows whether DoNotDisturb mode is switched on or not.

Value space of the result returned:
Active/Inactive

Example:
xStatus Conference DoNotDisturb
's Conference DoNotDisturb: Inactive
** end

xStatus Conference Line [n] Mode
Indicates whether the system is configured as private or shared line on CUCM.

Value space of the result returned:
Shared/Private

Example:
xStatus Conference Line Mode
's Conference Line 1 Mode: Private
** end
xStatus Conference Multipoint Mode
Shows how the Multipoint video conferences are handled.
Auto: The multipoint method available will be chosen automatically; if none are available the Multipoint Mode will automatically be set to Off. If both MultiWay and MultiSite are available, the MultiWay service takes priority over the built-in MultiSite.
Off: Multiparty conferences are not allowed.
MultiSite: Multiparty conferences are set up using the built-in MultiSite feature. If MultiSite is chosen when the MultiSite feature is not available, the Multipoint Mode will automatically be set to Off.
CUCMMediaResourceGroupList: Multiparty conferences (ad hoc conferences) are hosted by the CUCM configured conference bridge. This setting is provisioned by CUCM in a CUCM environment and should never be set manually by the user.
Value space of the result returned: Auto/CUCMMediaResourceGroupList/MultiSite/Off
Example:
```
xStatus Conference Multipoint Mode
*s Conference Multipoint Mode: "Auto"
** end
```

xStatus Conference Presentation CallId
Shows the identity of the system that sends the presentation.
Value space of the result returned: Integer
Example:
```
xStatus Conference Presentation CallId
*s Conference Presentation CallId: 0
** end
```

xStatus Conference Presentation LocalInstance \[n\] SendingMode
Shows whether a presentation source is shared locally or with a remote participant. There can be multiple local presentations which all have their own instance.
Value space of the result returned: LocalOnly/LocalRemote/Off
Example:
```
xStatus Conference Presentation LocalInstance 1 SendingMode
*s Conference Presentation LocalInstance 1 SendingMode: LocalOnly
** end
```

xStatus Conference Presentation LocalInstance \[n\] Source
Shows the SourceId for a current presentation. There can be multiple local presentations which all have their own instance.
Value space of the result returned: Integer
Example:
```
xStatus Conference Presentation LocalInstance 1 Source
*s Conference Presentation LocalInstance 1 Source: 1
** end
```

xStatus Conference Presentation Mode
Shows the status of the secondary video stream.
Value space of the result returned: On/Off
Example:
```
xStatus Conference Presentation Mode
*s Conference Presentation Mode: Off
** end
```
xStatus Conference SpeakerLock CallId
Shows the CallId for the participant locked as the prominent speaker in the conference.

Value space of the result returned:
Integer

Example:
xStatus Conference SpeakerLock CallId
  *s Conference SpeakerLock CallId: 0
  ** end

xStatus Conference SpeakerLock Mode
Shows whether a speaker lock is set or not.

Value space of the result returned:
On/Off

Example:
xStatus Conference SpeakerLock Mode
  *s Conference SpeakerLock Mode: Off
  ** end

Diagnostics status

xStatus Diagnostics
Shows the top level overview of the diagnostics. The example shows the status for an ongoing call. The identities of the call and channels are used when querying additional information.

xStatus Diagnostics Message [n] Description
A description of the current diagnostics alerts.

Value space of the result returned:
String

Example:
xStatus Diagnostics Message Description
  *s DiagnosticsResult Message 1 Description: "IP configuration incomplete"
  ** end

xStatus Diagnostics Message [n] Level
Returns information on the level of the diagnostics message.
Error: There is an error in the system. The system can still be used, but there can be some restrictions.
Warning: A problem is detected and a more specific report follows indicating the exact problem.
Critical: The warning level is critical. The system cannot be used.

Value space of the result returned:
Error/Warning/Critical

Example:
xStatus Diagnostics Message 4 Level
  *s Diagnostics Message 4 Level: Warning
  ** end
xStatus Diagnostics Message [n] References
Additional information on the diagnostics alert, if available.

Value space of the result returned:
String

Example:
* xStatus Diagnostics Message 10 References
  *s Diagnostics Message 10 References: "delay=190"
** end

xStatus Diagnostics Message [n] Type
Returns information on the results of the latest diagnostics on the system.

Value space of the result returned:
CAPFOperationState/CTLinstallation/CUCMVendorConfigurationFile/
  CallProtocolDualStackConfig/CallProtocolIPStackPlatformCompatibility/
  CallProtocolVcsProvisioningCompatibility/CameraId/CameraPairing/
  CameraStatus/CamerasDetected/ConfigurationFile/
  DefaultCallProtocolRegistered/EthernetDuplexMatches/FollowPresenterCameraConnection/
  H320GatewayStatus/H323GatekeeperStatus/HasValidReleaseKey/IpCameraStatus/
  IpV4Assignment/IpV6Assignment/IPv6Mtu/ISDNLinkCompatibility/ISDNLinkIpStack/
  ITLinstallation/InvalidSIPTransportConfig/LockDown/NetLinkStatus/NetSpeedAutoNegotiated/
  NTPstatus/OsdVideoOutput/OutputConnectorLocations/ProvisioningStatus/
  SIPProfileRegistration/SipProfileType/SelectedVideoInputSourceConnected/SipIceAndAnatConflict/
  TlsVerifyRequiredCerts/TurnBandwidth/UdpPortRangeViolation/ValidPasswords/
  VideoFromInternalCamera/VideoInputStability/SpeakerTrackFrontPanelMountedCorrectly/
  SpeakerTrackMicrophoneConnection/SpeakerTrackVideoInputs/
  SpeakerTrackEthernetConnection/ANATOnVCS/ECReferenceDelay/AudioPairingSNR/
  AudioInternalSpeakerDisabled/AbnormalCallTermination/HasActiveCallProtocol/
  SipOrH323ButNotBothEnabled/PresentationSourceSelection/SoftwareUpgrade/
  ContactInfoMismatch

Example:
* xStatus Diagnostics Message type
  *s Diagnostics Message 1 Type: InvalidAdminPassword
** end

H323 status

xStatus H323
Shows the top level overview of the H323 status.

xStatus H323 Gatekeeper Address
Displays the IP address of the gatekeeper where the system is registered.

Value space of the result returned:
String

Example:
* xStatus H323 Gatekeeper Address
  *s H323 Gatekeeper Address: "192.0.1.20"
** end

xStatus H323 Gatekeeper Port
Shows the port which is used when connecting to on the gatekeeper.

Value space of the result returned:
Integer

Example:
* xStatus H323 Gatekeeper Port
  *s H323 Gatekeeper Port: 1719
** end

xStatus H323 Gatekeeper Reason
Shows the reason for rejected registration.

Value space of the result returned:
String

Example:
* xStatus H323 Gatekeeper Reason
  *s H323 Gatekeeper Reason: ""
** end
xStatus H323 Gatekeeper Status
Shows the gatekeeper registration status.

Value space of the result returned:
- Required/Discovering/Discovered/Authenticating/Authenticated/Registering/Registered/
  Inactive/Rejected

Example:
```plaintext
xStatus H323 Gatekeeper Status
*s H323 Gatekeeper Status: Registered
** end
```

xStatus H323 Mode Reason
Shows whether there is a conflict between H.323 settings and xStatus H323 Mode Status.

- "": When H.323 is set to On and there is no conflict between H.323 Mode configuration and
  the rest of the system settings.
- "SIP is enabled": When H.323 Mode is set to On and SIP is enabled on a system that does not
  support the two simultaneously.
- "Not available": When a system does not support H.323.

Value space of the result returned:
- String

Example:
```plaintext
xStatus H323 Mode Reason
*s H323 Mode Reason: ""
** end
```

xStatus H323 Mode Status
Shows the status for H.323 registration.

- Enabled: Registration is enabled.
- Disabled: Registration is disable, because SIP is enabled.

Value space of the result returned:
- Enabled/Disabled

Example:
```plaintext
xStatus H323 Mode Status
*s H323 Mode Status: "Disabled"
** end
```

HttpFeedback status

xStatus HttpFeedback
Shows the top level overview of the HTTP status.

xStatus HttpFeedback [1..4] Expression [1..15]
Shows the feedback from the HTTP server. There can be up to 15 expressions for each URL.
See the xCommand HttpFeedback commands for more information.

Value space of the result returned:
- String

Example:
```plaintext
xStatus HttpFeedback 1 URL
** end
```
Network status

xStatus Network
Shows the top level overview of the network status.

xStatus Network 1 CDP Address
Returns the first network address of both receiving and sending devices.

Value space of the result returned:
String

Example:
```
xStatus Network CDP Address
*s Network 1 CDP Address: "192.0.1.20"
** end
```

xStatus Network 1 CDP Capabilities
Describes the functional capability for the switch in form of a device type. See documentation for CDP protocol for more information.

Value space of the result returned:
String

Example:
```
xStatus Network CDP Capabilities
*s Network 1 CDP Capabilities: "0x0029"
** end
```

xStatus Network 1 CDP Duplex
Indicates the status (duplex configuration) of the CDP broadcast interface. Used by network operators to diagnose connectivity problems between adjacent network elements.

Value space of the result returned:
String

Example:
```
xStatus Network CDP Duplex
*s Network 1 CDP Duplex: "Full"
** end
```

xStatus Network 1 CDP Platform
Returns the hardware platform name of the switch connected to the endpoint.

Value space of the result returned:
String

Example:
```
xStatus Network CDP Platform
*s Network 1 CDP Platform: "cisco WS-C3750X-48P"
** end
```

xStatus Network 1 CDP PortID
Returns the identification the switch uses of the port the endpoint is connected to.

Value space of the result returned:
String

Example:
```
xStatus Network CDP PortID
*s Network 1 CDP PortID: "GigabitEthernet1/0/23"
** end
```
xStatus Network 1 CDP PrimaryMgmtAddress
Returns the management address used to configure and monitor the switch the endpoint is connected to.

Value space of the result returned:
String

Example:
xStatus Network CDP PrimaryMgmtAddress
'*s Network 1 CDP PrimaryMgmtAddress: "10.1.1.2"
** end

xStatus Network 1 CDP SysName
Returns the SysName as configured in the switch the endpoint is connected to.

Value space of the result returned:
String

Example:
xStatus Network CDP SysName
'*s Network 1 CDP SysName: ""
** end

xStatus Network 1 CDP SysObjectID
Returns the SysObjectID as configured in the switch the endpoint is connected to.

Value space of the result returned:
String

Example:
xStatus Network CDP SysObjectID
'*s Network 1 CDP SysObjectID: ""
** end

xStatus Network 1 CDP Version
Returns information about the software release version the switch is running.

Value space of the result returned:
String

Example:
xStatus Network 1 CDP Version
** end

xStatus Network 1 CDP VoIPApplianceVlanID
Identifies the VLAN used for VoIP traffic from the endpoint to the switch. For more information see documentation of the IEEE 802.1Q protocol.

Value space of the result returned:
String

Example:
xStatus Network CDP VoIPApplianceVlanID
'*s Network 1 CDP VoIPApplianceVlanID: "300"
** end

xStatus Network 1 CDP VTPMgmtDomain
Returns the switch's configured VTP management domain name-string.

Value space of the result returned:
String

Example:
xStatus Network CDP VTPMgmtDomain
'*s Network 1 CDP VTPMgmtDomain: "anyplace"
** end
xStatus Network 1 DNS Domain Name
Shows the domain name.

Value space of the result returned:
String

Example:
```
xStatus Network 1 DNS Domain Name
*s Network 1 DNS Domain Name: "www.example.com www.example.int"
** end
```

xStatus Network 1 DNS Server [1..5] Address
Shows the IP address of the DNS server.

Value space of the result returned:
String

Example:
```
xStatus Network 1 DNS Server 1. Address
*s Network 1 DNS Server 1 Address: "192.0.2.60"
** end
```

xStatus Network 1 Ethernet MacAddress
Shows the MAC (Media Access Control) address for the Ethernet interface.

Value space of the result returned:
String

Example:
```
xStatus Network 1 Ethernet MacAddress
*s Network 1 Ethernet MacAddress: "00:50:60:02:FD:C7"
** end
```

xStatus Network 1 Ethernet Speed
Shows the Ethernet speed in Mbps. The speed can be in full-duplex or half-duplex.

Value space of the result returned:
10half/10full/100half/100full/1000full

Example:
```
xStatus Network 1 Ethernet Speed
*s Network 1 Ethernet Speed: "100full"
** end
```

xStatus Network 1 IPv4 Address
Shows the IPv4 address that uniquely identifies this system.

Value space of the result returned:
String

Example:
```
xStatus Network 1 IPv4 Address
*s Network 1 IPv4 Address: "192.0.2.149"
** end
```

xStatus Network 1 IPv4 Gateway
Shows the address of the IPv4 gateway.

Value space of the result returned:
String

Example:
```
xStatus Network 1 IPv4 Gateway
*s Network 1 IPv4 Gateway: "192.0.2.10"
** end
```
xStatus Network 1 IPv4 SubnetMask
Shows the subnet mask which determines which subnet an IPv4 address belongs to.

Value space of the result returned:
String

Example:
xStatus Network 1 IPv4 SubnetMask
  "s Network 1 IPv4 SubnetMask: "255.255.255.0"
  ** end

xStatus Network 1 IPv6 Address
Shows the IPv6 address that uniquely identifies this system.

Value space of the result returned:
String

Example:
xStatus Network 1 IPv6 Address
  "s Network 1 IPv6 Address: ""
  ** end

xStatus Network 1 IPv6 Gateway
Shows the address of the IPv6 gateway.

Value space of the result returned:
String

Example:
xStatus Network 1 IPv6 Gateway
  "s Network 1 IPv6 Gateway: ""
  ** end

xStatus Network 1 VLAN Voice VlanId
The feedback shows the VLAN Voice ID; or Off if the VLAN Voice Mode is not enabled.

Value space of the result returned:
Off/1..4094

Example:
xStatus Network 1 VLAN Voice VlanId
  "s Network 1 VLAN Voice VlanId: "Off"
  ** end
NetworkServices status

xStatus NetworkServices
Shows the top level overview of the network services status.

xStatus NetworkServices NTP CurrentAddress
Returns the address of the NTP server that is currently in use.

Value space of the result returned:
String

Example:
xStatus NetworkServices NTP CurrentAddress
`s NetworkServices NTP CurrentAddress: "123.254.15.121"
** end

xStatus NetworkServices NTP Server [n] Address
Returns the address of the NTP server(s) the codec is using.

Value space of the result returned:
String

Example:
xStatus NetworkServices NTP Address
`s NetworkServices NTP Address: "12.104.193.12 64.104.222.16 144.254.15.121"
** end

xStatus NetworkServices NTP Status
Returns the status of the endpoints synchronizing with the NTP server.
Unknown: State of the synchronization is unknown.
Synced: The system is in sync with the NTP server
Discarded: The NTP result has been discarded.

Value space of the result returned:
Unknown/Synced/Discarded

Example:
xStatus NetworkServices NTP Status
`s NetworkServices NTP Status: Synced
** end
Peripherals status

xStatus Peripherals
Shows the top level overview of the peripherals status.

xStatus Peripherals ConnectedDevice [n] HardwareInfo
Shows hardware information about connected device.

Value space of the result returned:
String

Example:
xStatus Peripherals ConnectedDevice 1007 HardwareInfo
*s Peripherals ConnectedDevice 1007 HardwareInfo: "1122330-0"
** end

xStatus Peripherals ConnectedDevice [n] ID
Shows the MAC-address of the connected device.

Value space of the result returned:
String

Example:
xStatus Peripherals ConnectedDevice 1007 ID
*s Peripherals ConnectedDevice 1007 ID: "00:10:20:20:be:21"
** end

xStatus Peripherals ConnectedDevice [n] Name
Shows the product name of connected device.

Value space of the result returned:
String

Example:
xStatus Peripherals ConnectedDevice 1007 Name
*s Peripherals ConnectedDevice 1007 Name: "Cisco TelePresence Touch"
** end

xStatus Peripherals ConnectedDevice [n] SoftwareInfo
Shows information of the software version running on the connected device.

Value space of the result returned:
String

Example:
xStatus Peripherals ConnectedDevice 1007 SoftwareInfo
*s Peripherals ConnectedDevice 1007 SoftwareInfo: "TII7.2.0"
** end

xStatus Peripherals ConnectedDevice [n] Status
Shows peripheral devices that are currently connected to the endpoint.

Value space of the result returned:
Connected/ResponseTimedOut

Example:
xStatus Peripherals ConnectedDevice 1001 Status
*s Peripherals ConnectedDevice 1001 Status: Connected
** end

xStatus Peripherals ConnectedDevice [n] Type
Shows the peripheral types that are connected to the endpoint.
Note: The value space Camera only shows Precision 60 cameras.

Value space of the result returned:
Byod/Camera/ControlSystem/ISDNLink/Other/SpeakerTrack/TouchPanel

Example:
xStatus Peripherals ConnectedDevice 1001 Type
*s Peripherals ConnectedDevice 1001 Type: TouchPanel
** end
xStatus Peripherals ConnectedDevice [n] UpgradeStatus
Shows the status of the previous software upgrade on the currently connected peripherals.

Value space of the result returned:
- Downloading
- Failed
- Installing
- InstallationReady
- None
- Succeeded
- Rebooting
- Retrying
- Aborted
- Paused

Example:
```
xStatus Peripherals ConnectedDevice 1001 UpgradeStatus
```
```
* Peripheral ConnectedDevice 1001 UpgradeStatus: None
```
** end

Provisioning status

xStatus Provisioning
Shows the top level overview of the provisioning status.

xStatus Provisioning Software Current CompletedAt
Shows date and time for when the current software upgrade was completed.

Value space of the result returned:
- String

Example:
```
xStatus Provisioning Software Current CompletedAt
```
```
* Provisioning Software Current CompletedAt: "2011-06-07T07:20:03Z"
```
** end

xStatus Provisioning Software Current URL
Shows the URL that the current software was uploaded from.

Value space of the result returned:
- String

Example:
```
xStatus Provisioning Software Current URL
```
```
* Provisioning Software Current URL: "http://.../s52020ce8_0_0.pkg"
```
** end

xStatus Provisioning Software Current VersionId
Shows the version ID of the current software.

Value space of the result returned:
- String

Example:
```
xStatus Provisioning Software Current VersionId
```
```
* Provisioning Software Current VersionId: "s52020ce8_0_0.pkg"
```
** end
xStatus Provisioning Software UpgradeStatus LastChange
Shows the date and time for the latest software upgrade.

Value space of the result returned:
String

Example:
xStatus Provisioning Software UpgradeStatus LastChange
"s Provisioning Software UpgradeStatus LastChange: "2011-06-07T07:20:03Z"
** end

xStatus Provisioning Software UpgradeStatus Message
Shows the system message for the software upgrade.

Value space of the result returned:
String

Example:
xStatus Provisioning Software UpgradeStatus Message
"s Provisioning Software UpgradeStatus Message: ""
** end

xStatus Provisioning Software UpgradeStatus Phase
Shows the phase of the software upgrade.

Value space of the result returned:
None/DownloadPending/FormingHierarchy/Downloading/DownloadPaused/DownloadDone/
Seeding/AboutToInstallUpgrade/Postponed/PeripheralsReady/UpgradingPeripherals/Installing/
InstallingPeripherals

Example:
xStatus Provisioning Software UpgradeStatus Phase
"s Provisioning Software UpgradeStatus Phase: None"
** end

xStatus Provisioning Software UpgradeStatus SessionId
Shows the ID of the session for the software upgrade.

Value space of the result returned:
String

Example:
xStatus Provisioning Software UpgradeStatus SessionId
"s Provisioning Software UpgradeStatus SessionId: ""
** end

xStatus Provisioning Software UpgradeStatus Status
Shows the status of the software upgrade.

Value space of the result returned:
None/InProgress/Failed/InstallationFailed/Succeeded

Example:
xStatus Provisioning Software UpgradeStatus Status
"s Provisioning Software UpgradeStatus Status: None"
** end

xStatus Provisioning Software UpgradeStatus URL
Shows the URL that the new software currently is being uploaded and installed from.

Value space of the result returned:
String

Example:
xStatus Provisioning Software UpgradeStatus URL
"s Provisioning Software UpgradeStatus URL: "http://.../s52020ce8_0_0.pkg"
** end
xStatus Provisioning Software UpgradeStatus VersionId
Shows the version ID of the software currently being uploaded and installed.

Value space of the result returned:
String

Example:
```plaintext
xStatus Provisioning Software UpgradeStatus VersionId
*s Provisioning Software UpgradeStatus VersionId: "s52010ce8_0_0.pkg"
** end
```

xStatus Provisioning Status
Shows the status of the provisioning.
Failed: The provisioning failed.
AuthenticationFailed: The authentication failed.
Provisioned: The endpoint is provisioned.
Idle: The provisioning is not active.
NeedConfig: The endpoint needs to be configured.
ConfigError: An error occurred during configuration.

Value space of the result returned:
Failed/AuthenticationFailed/Provisioned/Idle/NeedConfig/ConfigError

Example:
```plaintext
xStatus Provisioning Status
*s Provisioning Status: Provisioned
** end
```

Proximity status

xStatus Proximity
Shows the top level overview of the proximity status.

xStatus Proximity Services Availability
Shows whether proximity services are available on the endpoint.
Available: Proximity mode has been enabled with the command xConfiguration Proximity Mode and one or more of the proximity services have been enabled with xConfiguration Proximity Services commands.
Disabled: Proximity mode has been disabled with xConfiguration Proximity Mode, or none of the services have been enabled with the xConfiguration Proximity Services commands.
Deactivated: Proximity services have been deactivated with the command xCommand Proximity Services Deactivate.

Example:
```plaintext
xStatus Proximity Services Availability
*s Proximity Services Availability: Disabled
** end
```
Security status

xStatus Security
Shows the top level overview of the security status.

xStatus Security FIPS Mode
Shows the FIPS mode status.
Value space of the result returned:
On/Off
Example:
xStatus Security FIPS Mode
*s Security FIPS Mode: Off
** end

xStatus Security Persistency CallHistory
Shows whether call history logging is set to persistent or non-persistent mode. Persistent is the default mode.
Value space of the result returned:
NonPersistent/Persistent
Example:
xStatus Security Persistency CallHistory
*s Security Persistency CallHistory: Persistent
** end

xStatus Security Persistency Configurations
Shows whether the systems all configurations are set to persistent or non-persistent mode. Persistent is the default mode.
Value space of the result returned:
NonPersistent/Persistent
Example:
xStatus Security Persistency Configurations
*s Security Persistency Configurations: Persistent
** end

xStatus Security Persistency DHCP
Shows whether DHCP logging is set to persistent or non-persistent mode. Persistent is the default mode.
Value space of the result returned:
NonPersistent/Persistent
Example:
xStatus Security Persistency DHCP
*s Security Persistency DHCP: Persistent
** end

xStatus Security Persistency InternalLogging
Shows whether internal logging is set to persistent or non-persistent mode. Persistent is the default mode.
Value space of the result returned:
NonPersistent/Persistent
Example:
xStatus Security Persistency InternalLogging
*s Security Persistency InternalLogging: Persistent
** end

xStatus Security Persistency LocalPhonebook
Shows whether local phone book is set to persistent or non-persistent mode. Persistent is the default mode.
Value space of the result returned:
NonPersistent/Persistent
Example:
xStatus Security Persistency LocalPhonebook
*s Security Persistency LocalPhonebook: Persistent
** end
SIP status

xStatus SIP
Shows the top level overview of the SIP status.

xStatus SIP AlternateURI Alias [n] URI
Value space of the result returned:
String
Example:
xStatus SIP AlternateURI Alias
's SIP AlternateURI Alias URI: ""
** end

xStatus SIP AlternateURI Primary [n] URI
Value space of the result returned:
String
Example:
xStatus SIP AlternateURI Primary
's SIP AlternateURI Primary URI: ""
** end

xStatus SIP Authentication
Shows which authentication mechanism is used when registering to the SIP Proxy Server.
Digest: Uses the Digest access authentication method, as specified by RFC 2069.
NTLM: Uses the NTLM authentication method, which is a Microsoft authentication protocol.
Off: No authentication mechanism is used.
Value space of the result returned:
Digest/Off
Example:
xStatus SIP Authentication
's SIP Authentication: Off
** end

xStatus SIP CallForward DisplayName
Returns the URI that is displayed on the user interface for the forwarded call.
Value space of the result returned:
String
Example:
xStatus SIP CallForward DisplayName
's SIP CallForward DisplayName: ""
** end

xStatus SIP CallForward Mode
Indicates whether the call forward mode for SIP is set to on or off.
Value space of the result returned:
On/Off
Example:
xStatus SIP CallForward Mode
's SIP CallForward Mode: Off
** end

xStatus SIP CallForward URI
Indicates the address the incoming calls are directed to when call forward mode is set on.
Value space of the result returned:
String
Example:
xStatus SIP CallForward URI
's SIP CallForward URI: ""
** end
xStatus SIP Mailbox MessagesWaiting
Indicates how many new messages are in the mailbox.

**Value space of the result returned:**
Integer

**Example:**
xStatus SIP Mailbox MessagesWaiting
*s SIP Mailbox MessagesWaiting: 0
** end

xStatus SIP Mailbox URI
Returns the URI for your SIP mailbox.

**Value space of the result returned:**
String

**Example:**
xStatus SIP Mailbox URI
*s SIP Mailbox URI: "12345678"
** end

xStatus SIP Proxy [n] Address
Shows the address of the SIP Proxy that the system communicates with.

**Value space of the result returned:**
String

**Example:**
xStatus SIP Proxy 1 Address
*s SIP Proxy 1 Address: "192.0.2.50"
** end

xStatus SIP Proxy [n] Secure
Shows the encryption status of the signaling with the SIP Proxy server.

**Value space of the result returned:**
True/False

**Example:**
xStatus SIP Proxy 1 Secure
*s SIP Proxy 1 Secure: True
** end

xStatus SIP Proxy [n] Status
Shows the status of the communication between the endpoint and the SIP Proxy server.
Active: The communication between the endpoint and the SIP Proxy is active.
DNSFailed: The attempt to establish communication to the DNS server failed.
Off: There is no communication between the endpoint and the SIP Proxy.
Timeout: The attempt to establish communication to the SIP Proxy timed out.
UnableTCP: The system is unable to use TCP as the transport method.
UnableTLS: The system is unable to use TLS as the transport method.
Unknown: The status of the communication is not known.
AuthenticationFailed: Wrong user name or password.

**Value space of the result returned:**
Active/DNSFailed/Off/Timeout/UnableTCP/UnableTLS/Unknown/AuthenticationFailed

**Example:**
xStatus SIP Proxy 1 Status
*s SIP Proxy 1 Status: Active
** end
xStatus SIP Proxy [n] Verified
Shows whether or not the SSL certificate of the server that the video system / codec tries to register to is included in the codec’s trusted CA-list. The server is typically a Cisco VCS or CUCM.

True: The server’s SIP certificate is checked against the trusted CA-list on the codec and found valid. Additionally, the fully qualified domain name of the server matches the valid certificate.
False: A TLS connection is not set up because the SIP certificate verification failed or the domain name did not match. Note that the status also returns False when TLS is not used (xConfiguration SIP DefaultTransport not set to TLS) or certificate verification is switched off (SIP TlsVerify: Off. This setting is accessible through your products web interface).

Value space of the result returned: True/False

Example:
```
xStatus SIP Proxy 1 Verified
  *s SIP Proxy 1 Verified: False
  ** end
```

xStatus SIP Registration [n] Authentication
Shows which authentication mechanism is used when registering to the SIP Proxy Server.

Digest: Uses the Digest access authentication method, as specified by RFC 2069.
NTLM: Uses the NTLM authentication method, which is a Microsoft authentication protocol.
Off: No authentication mechanism is used.

Value space of the result returned: Digest/Off

Example:
```
xStatus SIP Registration 1 Authentication
  *s SIP Registration 1 Authentication: Off
  ** end
```

xStatus SIP Registration [n] Reason
Shows a message to explain the reason why the SIP registration failed.

Value space of the result returned: String

Example:
```
xStatus SIP Registration 1 Reason
  *s SIP Registration 1 Reason: "404 Not Found"
  ** end
```

xStatus SIP Registration [n] Status
Shows the status of the registration to the SIP Proxy Server.

Deregister: The system is in the process of de-registering to the SIP Proxy.
Failed: The system failed to register to the SIP Proxy.
Inactive: The system is not registered to any SIP Proxy.
Registered: The system is registered to the SIP Proxy.
Registering: The system is in the process of registering to the SIP Proxy.

Value space of the result returned: Deregister/Failed/Inactive/Registered/Registering

Example:
```
xStatus SIP Registration 1 Status
  *s SIP Registration 1 Status: Registered
  ** end
```

xStatus SIP Registration [n] URI
Shows the URI used for registration to the SIP Proxy server.

Value space of the result returned: String

Example:
```
xStatus SIP Registration 1 URI
  *s SIP Registration 1 URI: "firstname.lastname@company.com"
  ** end
```
xStatus SIP Secure
Shows the encryption status of the signaling with the SIP Proxy server.

Value space of the result returned:
True/False

Example:
xStatus SIP Secure
  *s SIP Secure: True
  ** end

xStatus SIP Verified
Shows whether or not the SSL certificate of the server that the video system / codec tries to register to is included in the codec's trusted CA-list. The server is typically a Cisco VCS or CUCM.

True: The server's SIP certificate is checked against the trusted CA-list on the codec and found valid. Additionally, the fully qualified domain name of the server matches the valid certificate.
False: A TLS connection is not set up because the SIP certificate verification failed or the domain name did not match. Note that the status also returns False when TLS is not used (xConfiguration SIP DefaultTransport not set to TLS) or certificate verification is switched off (SIP TlsVerify: Off. This setting is accessible through your products web interface).

Value space of the result returned:
True/False

Example:
xStatus SIP Verified
  *s SIP Verified: False
  ** end

Standby status

xStatus Standby
Shows the top level overview of the standby status.

xStatus Standby State
Shows whether the system is in standby mode or not.

Value space of the result returned:
Standby/Off

Example:
xStatus Standby State
  *s Standby State: Off
  ** end
SystemUnit status

xStatus SystemUnit
Shows the top level overview of the system unit status.

xStatus SystemUnit Hardware Module SerialNumber
Shows the serial number of the hardware module in the codec.
Value space of the result returned:
String
Example:
xStatus SystemUnit Hardware Module SerialNumber
*s SystemUnit Hardware Module SerialNumber: "F9AA99A00090"
** end

xStatus SystemUnit Hardware Monitoring Fan [n] Status
The feedback shows the speed (rpm) for the specified fan.
Value space of the result returned:
String
Example:
xStatus SystemUnit Hardware Monitoring Fan 1 Status
*s SystemUnit Hardware Monitoring Fan 1 Status: "locked on 1096 rpm"
** end

xStatus SystemUnit Hardware Temperature
Shows the current maximum temperature (degree Celsius) measured in the codec/system.
Value space of the result returned:
String
Example:
xStatus SystemUnit Hardware Temperature
*s SystemUnit Hardware Temperature: "64.0"
** end

xStatus SystemUnit Notifications Notification [n] Text
Lists text related to important system notifications. Notifications are issued e.g. when a system was rebooted because of a software upgrade, or when a factory reset has been performed. All the notifications can be removed from the list by issuing the xCommand SystemUnitNotifications RemoveAll command.
Value space of the result returned:
String
Example:
xStatus SystemUnit Notifications Notification 1 Text
*s SystemUnit Notifications Notification 1 Text: "OK"
** end

xStatus SystemUnit Notifications Notification [n] Type
Lists the system notification types. Notifications are issued e.g. when a system is rebooted because of a software upgrade, or when a factory reset is performed.
FactoryResetOK: This value is returned after a successful factory reset.
FactoryResetFailed: This value is returned after a failed factory reset attempt.
SoftwareUpgradeOK: This value is returned after a successful software upgrade.
SoftwareUpgradeFailed: This value is returned after a failed software upgrade attempt.
RebootRequired: This value is returned when a reboot is required.
Other: This value is returned for any other notifications.
All the notifications can be removed from the list by issuing the xCommand SystemUnitNotifications RemoveAll command.
Value space of the result returned:
FactoryResetOK, FactoryResetFailed, SoftwareUpgradeOK, SoftwareUpgradeFailed, RebootRequired, Other
Example:
xStatus SystemUnit Notifications Notification 1 Type
*s SystemUnit Notifications Notification 1 Type: SoftwareUpgradeOK
** end
**xStatus SystemUnit ProductId**
Shows the product identity.

*Value space of the result returned:*
String

*Example:*
xStatus SystemUnit ProductId
  *s SystemUnit ProductId: "Cisco TelePresence Codec SX80"
  ** end

**xStatus SystemUnit ProductPlatform**
Shows the product platform.

*Value space of the result returned:*
String

*Example:*
xStatus SystemUnit ProductPlatform
  *s SystemUnit ProductPlatform: "SX80"
  ** end

**xStatus SystemUnit ProductType**
Shows the product type.

*Value space of the result returned:*
String

*Example:*
xStatus SystemUnit ProductType
  *s SystemUnit ProductType: "Cisco Codec"
  ** end

**xStatus SystemUnit Software Name**
Shows the name of the software that is installed on the codec.

*Value space of the result returned:*
String

*Example:*
xStatus SystemUnit Software Name
  *s SystemUnit Software Name: "s52010"
  ** end

**xStatus SystemUnit Software OptionKeys Encryption**
Shows if the system has the option key installed that supports the encryption functionality.

*Value space of the result returned:*
False/True

*Example:*
xStatus SystemUnit Software OptionKeys Encryption
  *s SystemUnit Software OptionKeys Encryption: "true"
  ** end

**xStatus SystemUnit Software OptionKeys PremiumResolution**
Shows if the system has the option key installed that supports the PremiumResolution functionality.

*Value space of the result returned:*
False/True

*Example:*
xStatus SystemUnit Software OptionKeys PremiumResolution
  *s SystemUnit Software OptionKeys PremiumResolution: "true"
  ** end
xStatus SystemUnit Software OptionKeys RemoteMonitoring
Shows whether the system has the remote monitoring option key installed. Remote monitoring option key enables snapshots from the web interface, and from a remote paired Touch 10.

Value space of the result returned:
False/True

Example:
```bash
xStatus SystemUnit Software OptionKeys RemoteMonitoring
*s SystemUnit Software OptionKeys RemoteMonitoring: "true"
** end
```

xStatus SystemUnit Software ReleaseDate
Shows the release date of the software installed on the codec.

Value space of the result returned:
String

Example:
```bash
xStatus SystemUnit Software ReleaseDate
*s SystemUnit Software ReleaseDate: "2015-05-05"
** end
```

xStatus SystemUnit Software Version
Shows the software version installed on the codec.

Value space of the result returned:
String

Example:
```bash
xStatus SystemUnit Software Version
*s SystemUnit Software Version: "CE8.0.0"
** end
```

xStatus SystemUnit State NumberOfActiveCalls
Shows the number of active calls.

Value space of the result returned:
0..5

Example:
```bash
xStatus SystemUnit State NumberOfActiveCalls
*s SystemUnit State NumberOfActiveCalls: 0
** end
```

xStatus SystemUnit State NumberOfInProgressCalls
Shows the number of calls in progress.

Value space of the result returned:
0..5

Example:
```bash
xStatus SystemUnit State NumberOfInProgressCalls
*s SystemUnit State NumberOfInProgressCalls: 0
** end
```

xStatus SystemUnit State NumberOfSuspendedCalls
Shows the number of suspended calls.

Value space of the result returned:
0..5

Example:
```bash
xStatus SystemUnit State NumberOfSuspendedCalls
*s SystemUnit State NumberOfSuspendedCalls: 0
** end
```
xStatus SystemUnit Uptime
Show the number of seconds since the last restart of the codec.

Value space of the result returned:
Integer

Example:
```plaintext
xStatus SystemUnit Uptime
*s SystemUnit Uptime: 597095
** end
```

Time status

xStatus Time
Shows the top level overview of the time status.

xStatus Time SystemTime
Returns the date and time set on the system.

Value space of the result returned:
String

Example:
```plaintext
xStatus Time SystemTime
*s Time SystemTime: "2014-04-25T10:04:03Z"
** end
```
UserInterface status

xStatus UserInterface
Shows the top level overview of the video status.

xStatus UserInterface ContactInfo ContactMethod [n] Number
Returns the system's active contact information. This address is used to reach this endpoint.

Value space of the result returned:
String

Example:
```bash
xStatus UserInterface ContactInfo ContactMethod Number
*s UserInterface ContactInfo ContactMethod 1 Number: "12345678"
** end
```

xStatus UserInterface ContactInfo Name
Returns the system's active contact name. The result depends on which protocol, if any, the system is registered on. The automatically set contact name may have been overridden with the command xConfiguration UserInterface ContactInfo Type. This results in a diagnostics warning about contact mismatch.

Value space of the result returned:
String

Example:
```bash
xStatus UserInterface ContactInfo Name
*s UserInterface ContactInfo Name: "MySystem"
** end
```

Video status

xStatus Video
Shows the top level overview of the video status.

xStatus Video ActiveSpeaker PIPPosition
Shows the position of the active speaker’s image on the screen.

Value space of the result returned:
UpperLeft/UpperCenter/UpperRight/CenterLeft/CenterRight/LowerLeft/LowerRight

Example:
```bash
xStatus Video ActiveSpeaker PIPPosition
*s Video PIP ActiveSpeaker Position: UpperCenter
** end
```

xStatus Video Input
Shows the top level overview of the video input status.

xStatus Video Input Connector [n] Connected
Shows whether is something is connected to the specified connector. Not all connections can be detected.

Value space of the result returned:
False/True/Unknown

Example:
```bash
xStatus Video Input Connector 1 Connected
*s Video Input Connector 1 Connected: True
** end
```
xStatus Video Input Connector [n] SignalState
Shows the signal state for the specified input.
Unknown: The signal format is unknown.
OK: A signal is detected and the signal format is supported.
Unsupported: A signal is detected, but the signal format is not supported.

Value space of the result returned:
OK/Unknown/Unsupported

Example:
  xStatus Video Input Connector 1 SignalState
  *s Video Input Connector 1 SignalState: OK
  ** end

xStatus Video Input Connector [n] SourceId
Shows the identifier of the input source that the connector is associated with.

Value space of the result returned:
Integer

Example:
  xStatus Video Input Connector 1 SourceId
  *s Video Input Connector 1 SourceId: 1
  ** end

xStatus Video Input Connector [n] Type
Shows which connector type it is.

Value space of the result returned:
Composite/DVI/HDMI/Unknown/YC

Example:
  xStatus Video Input Connector 1 Type
  *s Video Input Connector 1 Type: HDMI
  ** end

xStatus Video Input MainVideoSource
Returns the local video input currently used as the main source. The main video source is set with the xConfiguration Video DefaultMainSource command.

Value space of the result returned:
Integer

Example:
  xStatus Video Input MainVideoSource
  *s Video Input MainVideoSource: 1
  ** end

xStatus Video Input Source [n] ConnectorId
Shows the identifier of the connector that is associated with the input source.

Value space of the result returned:
Integer

Example:
  xStatus Video Input Source 1 ConnectorId
  *s Video Input Source 1 ConnectorId: 1
  ** end

xStatus Video Input Source [n] FormatStatus
Shows the resolution format status for the video input source.

Value space of the result returned:
Ok/OutOfRange/NotFound/Interlaced/Error/Unknown

Example:
  xStatus Video Input Source 1 FormatStatus
  *s Video Input Source 1 Resolution FormatStatus: Ok
  ** end
xStatus Video Input Source [n] FormatType
Shows the resolution format type for the video input source.

Value space of the result returned:
Unknown/AnalogCVTBlanking/AnalogCVTReducedBlanking/AnalogGTFDefault/
AnalogGTFSecondary/AnalogDiscreteTiming/AnalogDMTBlanking/AnalogCEABlanking/Digital

Example:
```
xStatus Video Input Source 1 FormatType
*s Video Input Source 1 Resolution FormatType: Digital
** end
```

xStatus Video Input Source [n] MediaChannelId
For internal use only.

Value space of the result returned:
Integer

Example:
```
xStatus Video Input Source MediaChannelId
*s Video Input Source 1 MediaChannelId: 2
*s Video Input Source 2 MediaChannelId: 3
** end
```

xStatus Video Input Source [n] Resolution Height
Shows the resolution height (in pixels) for the video input source.

Value space of the result returned:
0..3000

Example:
```
xStatus Video Input Source 1 Resolution Height
*s Video Input Source 1 Resolution Height: 1080
** end
```

xStatus Video Input Source [n] Resolution RefreshRate
Shows the resolution refresh rate (Hz) for the video input source.

Value space of the result returned:
0..300

Example:
```
xStatus Video Input Source 1 Resolution RefreshRate
*s Video Input Source 1 Resolution RefreshRate: 50
** end
```

xStatus Video Input Source [n] Resolution Width
Shows the resolution width (in pixels) for the video input source.

Value space of the result returned:
0..4000

Example:
```
xStatus Video Input Source 1 Resolution Width
*s Video Input Source 1 Resolution Width: 1920
** end
```

xStatus Video Monitors
Returns the monitor layout mode.

Single: The same layout is shown on all monitors.
Dual: The layout is distributed on two monitors.
DualPresentationOnly: All participants in the call will be shown on the first monitor, while the
presentation (if any) will be shown on the second monitor.
Quadruple: The layout is distributed on four monitors, so that each remote participant and the
presentation will be shown on separate monitors.

Value space of the result returned:
Single/Dual/DualPresentationOnly/Triple/Quadruple

Example:
```
xStatus Video Monitors
*s Video Monitors: Single
** end
```
**xStatus Video Output**

Shows the top level overview of the video output status.

**xStatus Video Output Connector [n] Connected**

Describes whether a device (for example a display) is connected to the output connector or not. When a display enters standby mode, the endpoint may not be able to detect it. The connector status will then return False/Unknown even if the display is physically connected.

- **True**: A device is connected to the video output connector.
- **False**: Nothing is connected to the video output connector.

**Value space of the result returned:** True/False

**Example:**
```plaintext
xStatus Video Output Connector 1 Connected
`s Video Output Connector 1 Connected: True
```

**xStatus Video Output Connector [n] ConnectedDevice CEC DeviceType**

Shows the type of CEC enabled device connected to the HDMI output the codec has detected.

This information is only available when the device connected to the HDMI output has the CEC feature configured on and the codec has the configuration `xConfiguration Video Output Connector [n] CEC Mode` set to on.

**Value space of the result returned:** Unknown/TV/Reserved/Recorder/Tuner/Playback/Audio

**Example:**
```plaintext
xStatus Video Output Connector 1 ConnectedDevice CEC DeviceType
`s Video Output Connector 1 ConnectedDevice CEC DeviceType: TV
```

**xStatus Video Output Connector [n] ConnectedDevice CEC PowerControl**

Shows whether the codec is controlling the CEC enabled device connected to the HDMI output. This information is only available when the device connected to the HDMI output has the CEC feature configured on and the codec has the configuration `xConfiguration Video Output Connector CEC Mode` set to on.

**Value space of the result returned:** Unknown/Ok/In progress/Failed to power on/Failed to standby

**Example:**
```plaintext
xStatus Video Output Connector 1 ConnectedDevice CEC PowerControl
`s Video Output Connector 1 ConnectedDevice CEC PowerControl: Ok
```

**xStatus Video Output Connector [n] ConnectedDevice CEC PowerStatus**

Shows the state of the CEC enabled devise connected to the HDMI output. This information is only available when the device connected to the HDMI output has the CEC feature configured on and the codec has the configuration `xConfiguration Video Output Connector CEC Mode` set to on.

**Value space of the result returned:** Unknown/Ok/In progress/Failed to power on/Failed to standby

**Example:**
```plaintext
xStatus Video Output Connector 1 ConnectedDevice CEC PowerStatus
`s Video Output Connector 1 ConnectedDevice CEC PowerStatus: Ok
```

**xStatus Video Output Connector [n] ConnectedDevice Name**

Shows the name of the monitor connected to the HDMI port as defined in the monitors EDID.

**Value space of the result returned:** String

**Example:**
```plaintext
xStatus Video Output Connector 1 ConnectedDevice Name
`s Video Output Connector 1 ConnectedDevice Name: "G2420HDBL"
```
xStatus Video Output Connector [n] ConnectedDevice PreferredFormat

Shows the preferred input format of the monitor connected to the HDMI port as defined in the monitors EDID. This is not necessarily the format the codec is sending out.

Value space of the result returned:
String

Example:
xStatus Video Output Connector 1 ConnectedDevice PreferredFormat
  *s Video Output Connector 1 ConnectedDevice PreferredFormat: "1920x1080@60Hz"
** end

xStatus Video Output Connector [n] MonitorRole

Describes which video stream is shown on the device that is connected to the video output connector.
First/Second/Third: The role of the monitor in a multimonitor setup. In a singlemonitor setup, there is no difference between First, Second and Third.
PresentationOnly: Shows presentation video stream if active.
Recorder: Shows all participants, including the local main video. If active, shows also the presentation.

Value space of the result returned:
First/Second/Third/PresentationOnly/Recorder

Example:
xStatus Video Output Connector 1 MonitorRole
  *s Video Output Connector 1 MonitorRole: First
** end

xStatus Video Output Connector [n] Resolution Height

Shows the resolution height (in pixels) for the video output connector.

Value space of the result returned:
120..3000

Example:
xStatus Video Output Connector 1 Resolution Height
  *s Video Output Connector 1 Resolution Height: 1080
** end

xStatus Video Output Connector [n] Resolution RefreshRate

Shows the resolution refresh rate (Hz) for the video output connector.

Value space of the result returned:
1..300

Example:
xStatus Video Output Connector 1 Resolution RefreshRate
  *s Video Output Connector 1 Resolution RefreshRate: 60
** end

xStatus Video Output Connector [n] Resolution Width

Shows the resolution width (in pixels) for the video output connector.

Value space of the result returned:
176..4000

Example:
xStatus Video Output Connector 1 Resolution Width
  *s Video Output Connector 1 Resolution Width: 1920
** end

xStatus Video Output Connector [n] Type

Shows the type of connector.
HDMI: It is an HDMI connector.
DVI: It is an DVI connector.

Value space of the result returned:
HDMI/DVI

Example:
xStatus Video Output Connector 1 Type
  *s Video Output Connector 1 Type: HDMI
** end
**xStatus Video Presentation PIPPosition**
Shows the position of the presentation image on the screen.

*Value space of the result returned:*
UpperLeft/UpperCenter/UpperRight/CenterLeft/CenterRight/LowerLeft/LowerRight

*Example:*
```
xStatus Video Presentation PIPPosition
*s Video PIP Presentation Position: CenterLeft
** end
```

**xStatus Video Selfview FullscreenMode**
Shows whether selfview is set on full screen mode or not.

*Value space of the result returned:*
On/Off

*Example:*
```
xStatus Video Selfview FullscreenMode
*s Video Selfview FullscreenMode: Off
** end
```

**xStatus Video Selfview Mode**
Shows whether selfview mode is set on or not.

*Value space of the result returned:*
On/Off

*Example:*
```
xStatus Video Selfview Mode
*s Video Selfview Mode: Off
** end
```

**xStatus Video Selfview OnMonitorRole**
Identifies which monitor(s) contains the selfview, if present.

*Value space of the result returned:*
First/Second/Third/Fourth

*Example:*
```
xStatus Video Selfview OnMonitorRole
*s Video Selfview OnMonitorRole: First
** end
```

**xStatus Video Selfview PIPPosition**
Shows the position of the selfview image on the screen.

*Value space of the result returned:*
UpperLeft/UpperCenter/UpperRight/CenterLeft/CenterRight/LowerLeft/LowerRight

*Example:*
```
xStatus Video Selfview PIPPosition
*s Video Selfview PIPPosition: LowerRight
** end
```
Chapter 6

Appendices
About startup scripts

You can add one or more startup scripts to the codec. A startup script contains commands (xCommand) and configurations (xConfiguration) that will be executed as part of the start up procedure every time the codec boots. A few commands and configurations cannot be placed in a startup script, e.g. xCommand Boot.

Use the web interface of the codec to create and manage startup scripts.

Read more about the web interface and startup scripts in the Administrator guide for your product.
The SystemTools commands

**NOTE:** The systemtools commands are used for administrative control of the codec and are only available from a command line interface. Systemtools should not be used to program the codec.

Required parameters in angle brackets: `<text>`

Optional parameters in square brackets: `[text]`

To get an overview of the supported commands type “systemtools ?”.

Example:

```
systemtools ?
idefixversion
touchpanelversion
license
network
ntp
pairing
passwd
pki
rootsettings
securitysettings
securitystatus
selectsw
sudo
whoami
```

To see the usage of the commands add a question mark after the command.

Example:

```
systemtools network ?
usage: network ping <hostname> | traceroute <hostname> | netstat | addr | ifconfig
```

**systemtools idefixversion**
Returns the software version of a Touch controller that is connected to the codec.

**systemtools touchpanelversion**
Returns the software version the connected Touch controller should have. Used when upgrading to CE8.0 or later.

**systemtools license list**
Lists all the licenses for the codec.

**systemtools license show <name>**
Shows the content of a license file, defined by the name. `<name>`: The name of the license file.

**systemtools network ping <hostname>**
Network debug command. `<hostname>`: The IP address or URL of the host.

**systemtools network traceroute <hostname>**
Network debug command. `<hostname>`: The IP address or URL of the host.

**systemtools network netstat**
Network debug command.

**systemtools network addr**
Check the systems IP address.

**systemtools network ifconfig**
Network debug command.

**systemtools pairing unpair**
Remove association with Cisco TelePresence Touch controller.

**systemtools passwd**
Change the password for the logged in user.

**systemtools pki list**
Lists the codec certificate and CA list if they exist.

**systemtools pki delete <cert-name>**
Delete the codec certificate and CA list if they exist. `<cert-name>`: The name of the certificate.

**systemtools securitysettings jitc**
Set up security requirements so they meet JITC.
Set password and PIN polices enforced on the codec.

**systemtools securitysettings isjitc**
Check if the current settings are JTIC compliant.

**systemtools securitysettings default**
Revert to default security settings.

**systemtools securitysettings ask**
Query for the separate configurations. When issuing this command you can see each policy separately.

- Press enter to keep the current value.
- Enter a number and press enter to change the given policy.
- The default value “0” indicates no restrictions.

Max failed login attempts [0]?

- Number of failed logins until a user is set inactive.
Suspend-time after max failed login attempts (minutes) [0]?
  • Number of minutes the user is set inactive after maximum failed login attempts have been exceeded.

Max simultaneous sessions total [0]?
  • Maximum number of users that can be logged in simultaneously to web and maximum number of users that can be logged in simultaneous to ssh/Telnet.

Max simultaneous sessions per user [0]?
  • Maximum number of simultaneous sessions per user.

Number of passwords to remember [0]?
  • Number of previous passwords that the new password must differ from.

Number of PINs to remember [0]?
  • Number of previous PINs that the new PIN must differ from.

Maximum time between password renewals (days) [0]?
  • If the user has not changed the password within the renewal time the user will be set inactive.

Minimum time between password renewals (hours) [0]?
  • The user can only change password once within this limit.

Maximum time between PIN renewals (days) [0]?
  • If the user has not changed the PIN within the renewal time the user will be set inactive.

Minimum time between PIN renewals (hours) [0]?
  • The user can only change PIN once within this limit.

Maximum time between logins (days) [0]?
  • If the user has not logged in within this limit the user will be set inactive.

Max consecutive equal digits in PINs [0]?
  • Maximum consecutive equal digits in PINs.

Minimum number of digits in PINs [0]?
  • Minimum number of digits in PINs.

Max consecutive identical characters in passwords [0]?
  • Maximum consecutive identical characters in passwords.

Minimum number of characters in passwords [0]?
  • Minimum number of characters in passwords.

Maxmum number of characters in passwords [0]?
  • Maximum number of characters in passwords.

Minimum number of lower-case letters in passwords [0]?
  • Minimum number of lower-case letters in passwords.

Minimum number of upper-case letters in passwords [0]?
  • Minimum number of upper-case letters in passwords.

Minimum number of numerical characters in passwords [0]?
  • Minimum number of numerical characters in passwords.

Minimum number of special characters in passwords [0]?
  • Minimum number of special characters in passwords.

Minimum number of character groups in passwords [0]?
  • Minimum number of character groups in passwords.

Minimum number of character changed from previous password [0]?
  • Minimum number of character changed from previous password.

systemtools selectsw
Select which of the available software images to use. Changing the software image will restart the codec.
  No argument given: Lists the version of the available software images, and shows which one is active.
  image name: Swap to the software with this name.

systemtools sudo on
Changes the user role of the current session to the role of the specified user
  <username>: The name of the sudo user.
  <password>: The password for the sudo user.

systemtools sudo off
Reverts back to the user role of the signed in user.

systemtools whoami
Lists the name and id of the signed in user, and the user roles held by this user.

systemtools securitystatus
Shows the security status for the codec.
## About disconnect cause types

The following parameters are logged when a call is disconnected. The disconnect cause types are used in disconnect events (xEvent).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CauseValue</strong></td>
<td>Proprietary. We recommend the use of CauseType and CauseCode.</td>
</tr>
<tr>
<td><strong>CauseType</strong></td>
<td>Describes why the call was disconnected. The value space is { OtherLocal, LocalDisconnect, UnknownRemoteSite, LocalBusy, LocalReject, InsufficientSecurity, OtherRemote, RemoteDisconnect, RemoteBusy, RemoteRejected, RemoteNoAnswer, CallForwarded, NetworkRejected }</td>
</tr>
<tr>
<td><strong>CauseString</strong></td>
<td>Describes the Cause Code.</td>
</tr>
<tr>
<td><strong>CauseCode</strong></td>
<td>The disconnect Cause Codes are defined in SIP and Q.850.</td>
</tr>
<tr>
<td><strong>CauseOrigin</strong></td>
<td>SIP, Q.850, internal.</td>
</tr>
</tbody>
</table>

**Example:**

```plaintext
xEvent DisconnectEvent

*e CallDisconnect
  CauseValue: 1
  CauseType: "LocalDisconnect"
  CauseString: ""
  OrigCallDirection: "outgoing"
  RemoteURI: "firstname.lastname@company.com"
  CallId: 89
  CauseCode: 0
  CauseOrigin: SIP
** end
```
User documentation on the Cisco web site

User documentation for the Cisco TelePresence products is available at

http://www.cisco.com/go/telepresence/docs

Choose a product category in the right pane until you find the correct product. This is the path you have to follow:

TelePresence Integration Solutions >
Cisco TelePresence SX Series

Alternatively, use the following short-link to find the documentation:

http://www.cisco.com/go/sx-docs

The documents are organized in the following categories:

Install and Upgrade > Install and Upgrade Guides
- Installation guides: How to install the product
- Getting started guide: Initial configurations required to get the system up and running
- RCSI guide: Regulatory compliance and safety information

Maintain and Operate > Maintain and Operate Guides
- Getting started guide: Initial configurations required to get the system up and running
- Administrator guide: Information required to administer your product
- Administering CE Endpoints on CUCM: Tasks to perform to start using the product with the Cisco Unified Communications Manager (CUCM)

Maintain and Operate > End-User Guides
- User guides: How to use the product
- Quick reference guides: How to use the product

Reference Guides | Command references
- API reference guides: Reference guide for the Application Programmer Interface (API)

Reference Guides > Technical References
- CAD drawings: 2D CAD drawings with measurements

Troubleshoot and Alerts > Troubleshooting Guides
- Knowledge base articles: Brief articles that give advice on installation, interoperability, configuration, and other frequently asked questions

Configure > Configuration Guides
- TC Console user guide: How to use the TC Console application, which provides a graphical interface to the advanced customizable features of the codec.

Design > Design Guides
- Video conferencing room guidelines: General guidelines for room design and best practice
- Video conferencing room guidelines: Things to do to improve the perceived audio quality

Software Downloads, Release and General Information > Licensing Information
- Open source documentation: Licenses and notices for open source software used in this product

Software Downloads, Release and General Information > Release Notes
- Software release notes