Poly CCX Business Media Phones with OpenSIP

Getting Help
For more information about installing, configuring, and administering Poly/Polycom products or services, go to Polycom Support.

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

© 2020 Plantronics, Inc. All rights reserved. Poly, the propeller design, and the Poly logo are trademarks of Plantronics, Inc. All other trademarks are the property of their respective owners.
Contents

Before You Begin..............................................................................................................14
  Audience, Purpose, and Required Skills....................................................................14
  Related Poly and Partner Resources.........................................................................14

Getting Started.............................................................................................................15
  CCX Phone Hardware Overview................................................................................16
  Poly CCX 400 Business Media Phones Hardware.................................................16
  Poly CCX 500 Business Media Phones Hardware.................................................18
  Poly CCX 600 Business Media Phones Hardware.................................................20
  Poly CCX 700 Business Media Phones Hardware.................................................22
  LED Indicators on CCX Phones.............................................................................23
  Poly CCX Phones Model Numbers..........................................................................24
  Working with UC Software....................................................................................24
  UC Software Provisioning Methods.....................................................................24
  Record the Phone’s Version Information............................................................25
  Power On and Configure Your Phone...................................................................25

Required Ports.............................................................................................................26
  Ports Used on Poly Phones....................................................................................26

Configuring Security Options.....................................................................................28
  Administrator and User Passwords.................................................................28
  Change the Administrator Password on the Phone...........................................29
  Administrator and User Password Parameters.................................................29
  Setting a Security Banner on the Phone.............................................................30
  Web Configuration Utility Security Banner Parameters....................................30
  Locking the System Web Interface after Failed Login Attempts.........................30
  Web Configuration Utility Lock Parameters.....................................................30
  Disabling External Ports and Features.................................................................31
  Disable Unused Ports and Features Parameters.............................................32
  Visual Security Classification...............................................................................33
  Visual Security Classification Parameters.........................................................33
  Encryption............................................................................................................34
  Encrypting Configuration Files............................................................................34
  Configuration File Encryption Parameters.........................................................35
  Voice over Secure IP............................................................................................36
  VoSIP Parameter...............................................................................................36
<table>
<thead>
<tr>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microbrowser and Web Browser Parameters</td>
</tr>
<tr>
<td>Support for REST API</td>
</tr>
<tr>
<td>REST API Parameter</td>
</tr>
<tr>
<td>Soft Keys</td>
</tr>
<tr>
<td>Call State for Custom Soft Keys</td>
</tr>
<tr>
<td>Soft Key Parameters</td>
</tr>
<tr>
<td>Soft Key Customization Parameters</td>
</tr>
<tr>
<td>Disabling Default Soft Keys</td>
</tr>
<tr>
<td>Enhanced Feature Keys</td>
</tr>
<tr>
<td>Enhanced Feature Keys Parameters</td>
</tr>
<tr>
<td>Some Guidelines for Configuring Enhanced Feature Keys</td>
</tr>
<tr>
<td>Macro Definitions</td>
</tr>
<tr>
<td>Flexible Line Key Assignment</td>
</tr>
<tr>
<td>Flexible Line Keys Parameters</td>
</tr>
<tr>
<td>Assigning Busy Lamp Field (BLF) and Presence to Line Keys</td>
</tr>
<tr>
<td>Directories and Contacts</td>
</tr>
<tr>
<td>Local Contact Directory</td>
</tr>
<tr>
<td>Local Contact Directory Parameters</td>
</tr>
<tr>
<td>Maximum Capacity of the Local Contact Directory</td>
</tr>
<tr>
<td>Creating Per-Phone Directory Files</td>
</tr>
<tr>
<td>Speed Dials</td>
</tr>
<tr>
<td>Speed Dial Contacts Parameters</td>
</tr>
<tr>
<td>Corporate Directory</td>
</tr>
<tr>
<td>Securely Store LDAP Credentials</td>
</tr>
<tr>
<td>Corporate Directory Parameters</td>
</tr>
<tr>
<td>Call Lists</td>
</tr>
<tr>
<td>Call List Parameters</td>
</tr>
<tr>
<td>Call Log Elements and Attributes</td>
</tr>
<tr>
<td>Call Controls</td>
</tr>
<tr>
<td>Microphone Mute</td>
</tr>
<tr>
<td>Persistent Microphone Mute</td>
</tr>
<tr>
<td>Persistent Microphone Mute Parameter</td>
</tr>
<tr>
<td>Call Timers</td>
</tr>
<tr>
<td>Called Party Identification</td>
</tr>
<tr>
<td>Calling Party Identification Parameters</td>
</tr>
<tr>
<td>Connected Party Identification</td>
</tr>
<tr>
<td>Calling Party Identification</td>
</tr>
<tr>
<td>Calling Party Identification Parameters</td>
</tr>
<tr>
<td>Remote Party Caller ID from SIP Messages</td>
</tr>
<tr>
<td>Topic</td>
</tr>
<tr>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Multiple Line Registrations</td>
</tr>
<tr>
<td>Maximum Number of Registrations</td>
</tr>
<tr>
<td>Multiple Line Registrations Parameters</td>
</tr>
<tr>
<td>Multiple Line Keys Per Registration</td>
</tr>
<tr>
<td>Multiple Line Keys Per Registration Parameter</td>
</tr>
<tr>
<td>Multiple Call Appearances</td>
</tr>
<tr>
<td>Multiple Call Appearance Parameters</td>
</tr>
<tr>
<td>Flexible Call Appearances</td>
</tr>
<tr>
<td>Bridged Line Appearance</td>
</tr>
<tr>
<td>Bridged Line Appearance Signaling</td>
</tr>
<tr>
<td>Bridged Line Appearance Parameters</td>
</tr>
<tr>
<td>Voicemail</td>
</tr>
<tr>
<td>Voicemail Parameters</td>
</tr>
<tr>
<td>Local Call Recording</td>
</tr>
<tr>
<td>Local Call Recording Parameter</td>
</tr>
<tr>
<td>Centralized Call Recording</td>
</tr>
<tr>
<td>Centralized Call Recording Parameters</td>
</tr>
<tr>
<td>Busy Lamp Field (BLF)</td>
</tr>
<tr>
<td>BLF Icons</td>
</tr>
<tr>
<td>Busy Lamp Field Actions</td>
</tr>
<tr>
<td>BLF Feature Options</td>
</tr>
<tr>
<td>Key System Emulation</td>
</tr>
<tr>
<td>Key System Emulation Parameters</td>
</tr>
<tr>
<td>Configuring Key System Emulation</td>
</tr>
<tr>
<td>Instant Messaging</td>
</tr>
<tr>
<td>Instant Messaging Parameter</td>
</tr>
<tr>
<td>Local and Centralized Conference Calls</td>
</tr>
<tr>
<td>Local and Centralized Conference Call Parameters</td>
</tr>
<tr>
<td>Conference Management</td>
</tr>
<tr>
<td>Conference Management Parameter</td>
</tr>
<tr>
<td>Local Digit Map</td>
</tr>
<tr>
<td>Local Digit Maps Parameters</td>
</tr>
<tr>
<td>OpenSIP Digit Map</td>
</tr>
<tr>
<td>Generating Secondary Dial Tone with Digit Maps</td>
</tr>
<tr>
<td>Enhanced 911 (E.911)</td>
</tr>
<tr>
<td>Enhanced 911 (E.911) Parameters</td>
</tr>
<tr>
<td>MLPP for AS-SIP</td>
</tr>
<tr>
<td>Setting Call Precedence with a Digit Map</td>
</tr>
<tr>
<td>Preemption Behavior on Low Priority Calls</td>
</tr>
<tr>
<td>MLPP with Shared Lines</td>
</tr>
<tr>
<td>MLPP with Call Transfer</td>
</tr>
<tr>
<td>MLPP with Conference Calls</td>
</tr>
<tr>
<td>Section</td>
</tr>
<tr>
<td>---------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MLPPP with n-way Conference Calls</td>
</tr>
<tr>
<td>MLPP with AS-SIP Parameters</td>
</tr>
<tr>
<td>AS-SIP Namespace Parameter</td>
</tr>
<tr>
<td>International Dialing Prefix</td>
</tr>
<tr>
<td>International Dialing Prefix Parameters</td>
</tr>
<tr>
<td>Shared Lines</td>
</tr>
<tr>
<td>Shared Call Appearances</td>
</tr>
<tr>
<td>Shared Call Appearances Parameters</td>
</tr>
<tr>
<td>Private Hold on Shared Lines</td>
</tr>
<tr>
<td>Private Hold on Shared Lines Parameters</td>
</tr>
<tr>
<td>Intercom Calls</td>
</tr>
<tr>
<td>Creating a Custom Intercom Soft Key</td>
</tr>
<tr>
<td>Intercom Calls Parameters</td>
</tr>
<tr>
<td>Push-to-Talk</td>
</tr>
<tr>
<td>Push-to-Talk Parameters</td>
</tr>
<tr>
<td>Group Paging</td>
</tr>
<tr>
<td>Group Paging Parameters</td>
</tr>
<tr>
<td>SIP-B Automatic Call Distribution</td>
</tr>
<tr>
<td>SIP-B Automatic Call Distribution Parameters</td>
</tr>
<tr>
<td>User Profiles</td>
</tr>
<tr>
<td>User Profile Parameters</td>
</tr>
<tr>
<td>Remotely Logging Out Users</td>
</tr>
<tr>
<td>User Profile Authentication</td>
</tr>
<tr>
<td>User Profile Server Authentication</td>
</tr>
<tr>
<td>User Profile Phone Authentication</td>
</tr>
<tr>
<td>Network Configuration</td>
</tr>
<tr>
<td>Two-Way Active Measurement Protocol</td>
</tr>
<tr>
<td>TWAMP Limitations</td>
</tr>
<tr>
<td>Two-Way Active Measurement Protocol Configuration Parameters</td>
</tr>
<tr>
<td>3GPP Technical Specifications</td>
</tr>
<tr>
<td>3GPP Technical Specifications Parameters</td>
</tr>
<tr>
<td>Technical Report-069</td>
</tr>
<tr>
<td>TR-069 Parameters</td>
</tr>
<tr>
<td>Configuring TR-069</td>
</tr>
<tr>
<td>Map TR-106 Parameters to Poly Parameters</td>
</tr>
<tr>
<td>Map TR-104 Parameters to Poly Parameters</td>
</tr>
<tr>
<td>Supported TR-069 Remote Procedure Call (RPC) Methods</td>
</tr>
</tbody>
</table>
Third-Party Servers ........................................................................................................ 385

Microsoft Exchange Integration .................................................................................. 385
  Integrating with Microsoft Exchange .................................................................... 385
  Configuring the Microsoft Exchange Server ..................................................... 386

Ribbon Communications Server ................................................................................. 391
  Multiple Appearance Directory Number - Single Call Appearance (MADN-SCA) 391
  Global Address Book (GAB) .............................................................................. 395
  Personal Address Book (PAB) ............................................................................ 397
  Enhanced 911 (E.911) Location for Ribbon Communications ............................ 398
  Emergency Instant Messages ............................................................................ 399

BroadSoft BroadWorks Server .................................................................................. 400
  Authentication with BroadWorks Xtended Service Platform (XSP) Service Interface 400
  BroadWorks Call Decline Policy ....................................................................... 401
  Flexible Seating ............................................................................................... 402
  Executive-Assistant .......................................................................................... 403
  Enhanced Call Park ........................................................................................... 405
  BroadSoft Directory Support ............................................................................ 406
  Polycom BroadSoft UC-One Application ......................................................... 406
  BroadSoft UC-One Directory Parameters ....................................................... 409
  Enterprise Directory Default Search ................................................................. 410
  BroadSoft Server-Based Call Logs ................................................................... 410
  BroadSoft Server-Based Redial ......................................................................... 410
  Anonymous Call Rejection ................................................................................ 411
  Simultaneous Ring Personal ............................................................................. 412
  Line ID Blocking ............................................................................................... 412
  BroadWorks Anywhere ..................................................................................... 413
  BroadSoft Server-based Call Waiting ................................................................. 413
  Remote Office .................................................................................................... 413
  BroadSoft UC-One Credentials ........................................................................ 414
  BroadSoft Server-Based Call Forwarding .......................................................... 415
  Hoteling ............................................................................................................... 415
  Feature-Synchronized Automatic Call Distribution (ACD) ......................... 416

Configuring uaCSTA ............................................................................................... 418
  Enable uaCSTA as a Dedicated Line ................................................................. 419
  uaCSTA Parameters .......................................................................................... 420

Device Parameters ................................................................................................... 421

Changing Device Parameters .................................................................................. 421
  Types of Device Parameters ............................................................................. 422
## Configuration Parameters

- Quick Setup Soft Key Parameter
- Per-Registration Call Parameters
- Remote Packet Capture Parameters
- Per-Registration Dial Plan Parameters
- Local Contact Directory File Size Parameters
- Parameter Elements for the Local Contact Directory
- Feature Activation and Deactivation Parameters
- HTTPD Web Server Parameters
- Feature License Parameter
- Chord Parameters
- Message Waiting Parameters
- Ethernet Interface MTU Parameters
- Presence Parameters
- Provisioning Parameters
- Configuration Request Parameter
- User Preferences Parameters
- Upgrade Parameters
- Voice Parameters
  - Acoustic Echo Suppression (AES) Parameter
  - Comfort Noise Parameters
  - Voice Jitter Buffer Parameters
  - Digital Gain Parameters
- SDP Parameters
- Download Location Parameter for Language Files
- XML Streaming Protocol Parameters
- Session Header Parameters
Before You Begin

Topics:

• Audience, Purpose, and Required Skills
• Related Poly and Partner Resources

This Poly CCX Business Media Phones with OpenSIP User Guide contains overview information for navigating and performing tasks on Poly CCX phones.

This user guide contains information for the following Poly products and accessories:

• CCX 400 business media phones
• CCX 500 business media phones
• CCX 600 business media phones
• CCX 700 business media phones
• Polycom EagleEye Mini USB camera

Audience, Purpose, and Required Skills

This guide is intended for beginning users, as well as intermediate and advanced users who want to learn more about the features available with CCX phones.

Related Poly and Partner Resources

See the following sites for information related to this product.

• The Poly Online Support Center is the entry point to online product, service, and solution support information including Video Tutorials, Documents & Software, Knowledge Base, Community Discussions, Poly University, and additional services.

• The Polycom Document Library provides support documentation for active products, services, and solutions. The documentation displays in responsive HTML5 format so that you can easily access and view installation, configuration, or administration content from any online device.

• The Polycom Community provides access to the latest developer and support information. Create an account to access Poly support personnel and participate in developer and support forums. You can find the latest information on hardware, software, and partner solutions topics, share ideas, and solve problems with your colleagues.

• The Polycom Partner Network are industry leaders who natively integrate the Poly standards-based RealPresence Platform with their customers' current UC infrastructures, making it easy for you to communicate face-to-face with the applications and devices you use every day.

• The Polycom Collaboration Services help your business succeed and get the most out of your investment through the benefits of collaboration.
Getting Started

Topics:

- CCX Phone Hardware Overview
- Working with UC Software
- Power On and Configure Your Phone

Before you use your phone, familiarize yourself with its features and user interface.

Note: As you read this guide, remember that your system administrator configures certain features or your network environment may determine them. As a result, some features may not be enabled or may operate differently on your device. Additionally, the examples and graphics in this guide may not directly reflect available options or what your device screen displays.
CCX Phone Hardware Overview

The following topics describe the hardware features for Poly CCX business media phones.

Poly CCX 400 Business Media Phones Hardware

The following figure displays the hardware features on Poly CCX 400 business media phones. The table lists each feature numbered in the figure.

Figure 1: Poly CCX 400 phone hardware features

<table>
<thead>
<tr>
<th>Reference Number</th>
<th>Feature</th>
<th>Feature Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reversible tab</td>
<td>Secures the handset in the cradle when you position your phone stand at a high angle. To secure the handset, remove the tab, reverse it so the protrusion points up, and reinsert it. Position the handset so the protrusion on the tab fits into the slot on the handset.</td>
</tr>
<tr>
<td>2</td>
<td>Message waiting indicator</td>
<td>Flashes red to indicate you have an incoming call or a new message.</td>
</tr>
<tr>
<td>Reference Number</td>
<td>Feature</td>
<td>Feature Description</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------</td>
<td>--------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>3</td>
<td>Touchscreen</td>
<td>Enables you to select items and navigate menus on the touch-sensitive screen.</td>
</tr>
<tr>
<td>4,5</td>
<td>Volume keys</td>
<td>Adjusts the volume of the handset, headset, speaker, and ringer.</td>
</tr>
<tr>
<td>6</td>
<td>USB port(s)</td>
<td>CCX 400 phones contain one USB-A port. Enables you to attach a USB flash drive or USB headset.</td>
</tr>
<tr>
<td>7</td>
<td>Headset indicator</td>
<td>Displays when the phone is powered on. The icon glows green or blue when a headset is connected and activated.</td>
</tr>
<tr>
<td>8</td>
<td>Speakerphone indicator</td>
<td>Displays when the phone is powered on. The icon glows green when activated.</td>
</tr>
<tr>
<td>9</td>
<td>Speaker</td>
<td>Provides ringer and speakerphone audio output.</td>
</tr>
<tr>
<td>10</td>
<td>Mute key</td>
<td>Mutes local audio during calls and conferences.</td>
</tr>
</tbody>
</table>
Poly CCX 500 Business Media Phones Hardware

The following figure displays the hardware features on Poly CCX 500 business media phones with OpenSIP. The table lists each feature numbered in the figure.

Figure 2: Poly CCX 500 phone hardware features

<table>
<thead>
<tr>
<th>Reference Number</th>
<th>Feature</th>
<th>Feature Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reversible tab</td>
<td>Secures the handset in the cradle when you position your phone stand at a high angle. To secure the handset, remove the tab, reverse it so the protrusion points up, and reinsert it. Position the handset so the protrusion on the tab fits into the slot on the handset.</td>
</tr>
<tr>
<td>2</td>
<td>Message waiting indicator</td>
<td>Flashes red to indicate you have an incoming call or new messages.</td>
</tr>
<tr>
<td>3</td>
<td>Application Command</td>
<td>Displays when the phone is powered on and the service is available. Icon may vary.</td>
</tr>
<tr>
<td>4</td>
<td>Touchscreen</td>
<td>Enables you to select items and navigate menus on the touch-sensitive screen.</td>
</tr>
<tr>
<td>Reference Number</td>
<td>Feature</td>
<td>Feature Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>5,6</td>
<td>Volume keys</td>
<td>Adjusts the volume of the handset, headset, speaker, and ringer.</td>
</tr>
<tr>
<td>7</td>
<td>USB port(s)</td>
<td>Poly CCX 500 phones include one USB-A port and one USB-C port. Enables you to attach a USB flash drive or USB headset.</td>
</tr>
<tr>
<td>8</td>
<td>Headset indicator</td>
<td>Displays when the phone is powered on. The icon glows green or blue when a headset is connected and activated.</td>
</tr>
<tr>
<td>9</td>
<td>Speakerphone indicator</td>
<td>Displays when the phone is powered on. The icon glows green when activated.</td>
</tr>
<tr>
<td>10</td>
<td>Speaker</td>
<td>Provides ringer and speakerphone audio output.</td>
</tr>
<tr>
<td>11</td>
<td>Mute key</td>
<td>Mutes local audio during calls and conferences.</td>
</tr>
</tbody>
</table>
Poly CCX 600 Business Media Phones Hardware

The following figure displays the hardware features on Poly CCX 600 business media phones. The table lists each feature numbered in the figure.

**Figure 3: Poly CCX 600 Phone Hardware Features**

---

**Poly CCX 600 Phone Hardware Feature Descriptions**

<table>
<thead>
<tr>
<th>Reference Number</th>
<th>Feature</th>
<th>Feature Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reversible tab</td>
<td>Secures the handset in the cradle when you position your phone stand at a high angle. To secure the handset, remove the tab, reverse it so the protrusion points up, and reinsert it. Position the handset so the protrusion on the tab fits into the slot on the handset.</td>
</tr>
<tr>
<td>2</td>
<td>Touchscreen</td>
<td>Enables you to select items and navigate menus on the touch-sensitive screen. Tap the screen to select and highlight screen items. To scroll, touch the screen, and swipe your finger up, down, right, or left.</td>
</tr>
<tr>
<td>3</td>
<td>Camera Port</td>
<td>Enables you to attach an optional Polycom EagleEye Mini USB camera.</td>
</tr>
<tr>
<td>Reference Number</td>
<td>Feature</td>
<td>Feature Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>4</td>
<td>Message waiting indicator</td>
<td>Flashes red to indicate you have an incoming call or new messages.</td>
</tr>
<tr>
<td>5</td>
<td>Application Command</td>
<td>Displays when the phone is powered on and the service is available. Icon may vary.</td>
</tr>
<tr>
<td>6,7</td>
<td>Volume keys</td>
<td>Adjusts the volume of the handset, headset, speaker, and ringer.</td>
</tr>
<tr>
<td>8</td>
<td>USB port(s)</td>
<td>One USB-A port and one USB-C port. Enables you to attach a USB flash drive or USB headset. You can connect an EagleEye Mini USB camera to the USB-A port.</td>
</tr>
<tr>
<td>9</td>
<td>Headset indicator</td>
<td>Displays when the phone is powered on. The icon glows green or blue when a headset is connected and activated.</td>
</tr>
<tr>
<td>10</td>
<td>Speakerphone indicator</td>
<td>Displays when the phone is powered on. The icon glows green when activated.</td>
</tr>
<tr>
<td>11</td>
<td>Speaker</td>
<td>Provides ringer and speakerphone audio output.</td>
</tr>
<tr>
<td>12</td>
<td>Mute key</td>
<td>Mutes local audio during calls and conferences.</td>
</tr>
</tbody>
</table>
Poly CCX 700 Business Media Phones Hardware

The following figure displays the hardware features on Poly CCX 700 business media phones. The table lists each feature numbered in the figure.

Figure 4: Poly CCX 700 Phone Hardware Features

Poly CCX 700 Phone Hardware Feature Descriptions

<table>
<thead>
<tr>
<th>Reference Number</th>
<th>Feature</th>
<th>Feature Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reversible tab</td>
<td>Secures the handset in the cradle when you position your phone stand at a high angle. To secure the handset, remove the tab, reverse it so the protrusion points up, and reinsert it. Position the handset so the protrusion on the tab fits into the slot on the handset.</td>
</tr>
<tr>
<td>2</td>
<td>Privacy Shutter</td>
<td>Allows you to cover the camera lens.</td>
</tr>
<tr>
<td>3</td>
<td>Built-in Camera</td>
<td>Enables you to send video during a video-enabled call.</td>
</tr>
<tr>
<td>4</td>
<td>Camera LED</td>
<td>Turns green when the camera is on and sending video. Turns amber when you are not sending video.</td>
</tr>
<tr>
<td>Reference Number</td>
<td>Feature</td>
<td>Feature Description</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
<td>---------------------</td>
</tr>
<tr>
<td>5</td>
<td>Touchscreen</td>
<td>Enables you to select items and navigate menus on the touch-sensitive screen. Tap the screen to select and highlight screen items. To scroll, touch the screen, and swipe your finger up, down, right, or left.</td>
</tr>
<tr>
<td>6</td>
<td>Message waiting indicator</td>
<td>Flashes red to indicate when you have new messages.</td>
</tr>
<tr>
<td>7</td>
<td>Application Command*</td>
<td>Displays when the phone is powered on and the service is available. Icon may vary.</td>
</tr>
<tr>
<td>8,9</td>
<td>Volume keys</td>
<td>Adjusts the volume of the handset, headset, speaker, and ringer.</td>
</tr>
<tr>
<td>10</td>
<td>Headset indicator</td>
<td>Displays when the phone is powered on. The icon glows green or blue when a headset is connected and activated.</td>
</tr>
<tr>
<td>11</td>
<td>Speakerphone indicator</td>
<td>Displays when the phone is powered on. The icon glows green when activated.</td>
</tr>
<tr>
<td>12</td>
<td>USB port(s)</td>
<td>CCX 700 phones contain one USB-A port and one USB-C port. Enables you to attach a USB flash drive or USB headset.</td>
</tr>
<tr>
<td>13</td>
<td>Speaker</td>
<td>Provides ringer and speakerphone audio output.</td>
</tr>
<tr>
<td>14</td>
<td>Mute key</td>
<td>Mutes local audio during calls and conferences.</td>
</tr>
</tbody>
</table>

### LED Indicators on CCX Phones
The LEDs on Poly CCX phones indicate phone and accessory states.

#### CCX Phones LED Descriptions

<table>
<thead>
<tr>
<th>LED</th>
<th>Color State and Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speakerphone</td>
<td>Off – Phone is in sleep mode. White – Speakerphone is off, but the phone is active. Green – Speakerphone is on.</td>
</tr>
<tr>
<td>Headset</td>
<td>Off – Phone is in sleep mode. White – No headset is active. Green – An electronic hookswitch (EHS) headset is connected and activated. Blue – A USB or Bluetooth headset is connected and activated.</td>
</tr>
</tbody>
</table>
### LED Color State and Description

<table>
<thead>
<tr>
<th>LED</th>
<th>Color State and Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mute</td>
<td>Off – No call is active.</td>
</tr>
<tr>
<td></td>
<td>Red – The phone microphone is muted.</td>
</tr>
<tr>
<td>Message Waiting Indicator (MWI)</td>
<td>Off – No new messages are available.</td>
</tr>
<tr>
<td></td>
<td>Blinking Red – New or unread voicemails are available, incoming call and missed call.</td>
</tr>
</tbody>
</table>

### Poly CCX Phones Model Numbers

The following table lists the product names, model numbers for Poly CCX business media phones.

<table>
<thead>
<tr>
<th>Product Name</th>
<th>Model Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poly CCX 400 business media phone</td>
<td>3111-49700-001</td>
</tr>
<tr>
<td>Poly CCX 500 business media phone</td>
<td>3111-49710-001</td>
</tr>
<tr>
<td>Poly CCX 600 business media phone</td>
<td>3111-49770-001</td>
</tr>
<tr>
<td>Poly CCX 700 business media phone</td>
<td>3111-49740-001</td>
</tr>
</tbody>
</table>

### Working with UC Software

Poly phones come installed with updater software that resides in the flash memory of the phone. When you boot up or reboot the phone, the updater automatically updates, downloads, and installs new software versions or configuration files as needed, based on the server or phone settings.

### UC Software Provisioning Methods

Poly provides several methods to provision phones and configure phone features. The method you use depends on the number of phones in your deployment, the phone model(s), and how you want to apply features and settings.

You can use multiple methods simultaneously to provision and configure features. There is a priority among the methods that impacts your phone deployment when you use multiple methods simultaneously. If there is a discrepancy among multiple provisioning methods or configuration settings, the Poly phone uses the setting set with the higher-priority method based on the following hierarchy:

1. Quick setup
2. Local interface (the phone menu)
3. System web interface (Web Configuration Utility)
4. USB
5. Polycom RealPresence Resource Manager
6. Centralized provisioning
7. Default phone values
For example, when you provision the phones using a provisioning server and subsequently apply settings using the system web interface, the system web interface setting overrides any duplicate settings you set from the provisioning server. Likewise, any settings set from the local interface override any duplicate settings you set using the system web interface.

For more information on provisioning phones, see the Poly CCX Business Media Phones Provisioning Guide.

Record the Phone’s Version Information
In case you need to contact technical support, record the following information for future reference:

- Phone model(s)
- Updater version
- UC software version
- Partner platform

Power On and Configure Your Phone
After powering on your phone for the first time, complete the phone setup wizard.

Procedure
1. Enter the default admin password 456.
2. Enter a new admin password and verify the password.
3. Read and accept the EULA.
4. Choose a system language.
   The default system language is English (United States).
5. Choose a timezone.
6. Choose the Generic base profile.
   The phone completes the startup process.
## Required Ports

### Topics:
- [Ports Used on Poly Phones](#)

Poly phones require certain network ports.

## Ports Used on Poly Phones

The following table lists the ports used by Polycom UC Software.

RTP and RTCP can use any even-numbered port between 2222 and 2269. Configure ports by setting `tcpIpApp.port.rtp.mediaPortRangeStart`.

### Ports Used by Poly Phones

<table>
<thead>
<tr>
<th>Port Number</th>
<th>Protocol</th>
<th>Outgoing</th>
<th>Incoming</th>
<th>UDP or TCP</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>FTP</td>
<td>Provisioning, Logs</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>22</td>
<td>SSH</td>
<td>Admin</td>
<td>Admin</td>
<td>TCP</td>
</tr>
<tr>
<td>23</td>
<td>Telnet</td>
<td>Admin</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>53</td>
<td>DNS</td>
<td></td>
<td></td>
<td>UDP</td>
</tr>
<tr>
<td>67</td>
<td>DHCP</td>
<td>Server</td>
<td></td>
<td>UDP</td>
</tr>
<tr>
<td>68</td>
<td>DHCP</td>
<td>Client</td>
<td></td>
<td>UDP</td>
</tr>
<tr>
<td>69</td>
<td>TFTP</td>
<td>Provisioning, Logs</td>
<td></td>
<td>UDP</td>
</tr>
<tr>
<td>80</td>
<td>HTTP</td>
<td>Provisioning, Logs, Pull Web interface, Poll</td>
<td>TCP</td>
<td></td>
</tr>
<tr>
<td>123</td>
<td>NTP</td>
<td>Time Server</td>
<td></td>
<td>UDP</td>
</tr>
<tr>
<td>389</td>
<td>LDAP</td>
<td></td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>443</td>
<td>HTTPS</td>
<td>Provisioning, Logs</td>
<td>HTTP Pull Web interface, HTTP Push</td>
<td>TCP</td>
</tr>
<tr>
<td>514</td>
<td>Syslog</td>
<td>Logs</td>
<td></td>
<td>UDP</td>
</tr>
<tr>
<td>636</td>
<td>LDAP</td>
<td>Logs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1468</td>
<td>Syslog</td>
<td>Logs</td>
<td></td>
<td>TCP</td>
</tr>
<tr>
<td>2222</td>
<td>RTP</td>
<td>Media Packets</td>
<td>Media Packets</td>
<td></td>
</tr>
<tr>
<td>Port Number</td>
<td>Protocol</td>
<td>Outgoing</td>
<td>Incoming</td>
<td>UDP or TCP</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------------</td>
<td>---------------------------</td>
<td>---------------------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>2223</td>
<td>RTCP</td>
<td>Media Packet Statistics</td>
<td>Media Packet Statistics</td>
<td></td>
</tr>
<tr>
<td>5060</td>
<td>SIP</td>
<td>SIP signaling</td>
<td>SIP signaling</td>
<td>TCP and UDP</td>
</tr>
<tr>
<td>5061</td>
<td>SIP over TLS</td>
<td>Secure signaling</td>
<td>Secure signaling</td>
<td>TCP</td>
</tr>
</tbody>
</table>
Configuring Security Options

Topics:

- Administrator and User Passwords
- Setting a Security Banner on the Phone
- Locking the System Web Interface after Failed Login Attempts
- Disabling External Ports and Features
- Visual Security Classification
- Encryption
- Voice over Secure IP
- Securing Phone Calls with SRTP
- Enabling Users to Lock Phones
- Locking the Basic Settings Menu
- Secondary Port Link Status Report
- USB Port Parameters
- 802.1X Authentication
- Simple Certificate Enrollment Protocol

Optimize security settings, such as changing the passwords for the phone, enabling users to lock their phones, and blocking administrator functions from phone users.

Administrator and User Passwords

You can change the default administrator and user passwords.

Poly strongly recommends that you change the default password. The default administrator password enables administrators to access advanced settings menu on the phone menu and to log in to a phone’s Web Configuration Utility as an administrator.

You can change the default password using any of the following methods:

- The pop-up prompt when the phone first registers
- Phone menu
- System web interface (Web Configuration Utility)
- Use the parameter reg.1.auth.password

You must have a user or administrator password before you can access certain menu options on the phone and in the system web interface. You can use the following default passwords to access menu options on the phone and to access the system web interface:

- Administrator password: 456
- User password: 123
You can use an administrator password where a user password is required to see all the user options. While you can use the user password where the administrator password is required, the phone displays a limited set of menu options. Note that the system web interface displays different features and options depending on which password you use.

**Change the Administrator Password on the Phone**

You set the administrator password during the initial system setup. You can’t use the default administrator password, 456, on the phone.

**Procedure**
1. On the phone, go to Settings > Advanced, and enter the current administrator password.
2. Select Administration Settings > Change Admin Password.
3. Enter the default password, enter a new password, and confirm the new password.
4. Enter the current password, enter a new password, and confirm the new password.

**Administrator and User Password Parameters**

Use the following parameters to set the administrator and user password and configure password settings.

sec.pwd.length.admin
- The minimum character length for administrator passwords changed using the phone. Use 0 to allow null passwords.
  - 1 (default)
  - 0 -32
  - Change causes system to restart or reboot.

sec.pwd.length.user
- The minimum character length for user passwords changed using the phone. Use 0 to allow null passwords.
  - 2 (default)
  - 0 -32
  - Change causes system to restart or reboot.

up.echoPasswordDigits
- 1 (default) The phone briefly displays password characters before masking them with an asterisk.
  - 0 - The phone displays only asterisks for the password characters.

device.auth.localAdminPassword
- Specify a local administrator password.
  - 0 - 32 characters
You must use this parameter with: `device.auth.localAdminPassword.set="1"

device.auth.localAdminPassword.set
0 (default) - Disables overwriting the local admin password when provisioning using a configuration file.
1 - Enables overwriting the local admin password when provisioning using a configuration file.

### Setting a Security Banner on the Phone
Configure a custom text message to display on the phone's system web interface.

**Web Configuration Utility Security Banner Parameters**
The following list includes the parameters of the web user interface for security banner parameters.

- **feature.webSecurityBanner.enabled**
  0 (default) - No security banner message displays on the phone's web user interface.
  1 - A security banner with the configured message displays phone's web user interface. Use `feature.webSecurityBanner.msg` to configure the message.

- **feature.webSecurityBanner.msg**
  Customize the text in security banner.
  "This is default text for the security log-on banner" (default) - This text displays because the security log-on banner has been enabled and the custom text to be displayed in the security log-on banner has not been configured.
  2000 characters (maximum)

### Locking the System Web Interface after Failed Login Attempts
Set the maximum number of login attempts allowed and the system lockout time.

For security, enable system lock. Set the maximum number of login attempts before the system locks and the system lockout time.

**Web Configuration Utility Lock Parameters**
Use the following parameters to configure how the Web Configuration Utility will behave after failed login attempts.

- **httpd.cfg.lockWebUI.enable**
  1 (default) - Enable the Web Configuration Login Lock feature.
  0 - Disable the Web Configuration Login Lock feature.
**httpd.cfg.lockWebUI.lockOutDuration**

60 seconds (default) - The period of time the user is locked out of the Web Configuration Utility. The user can try logging in again after this time.

60 - 300 seconds

The lock-out timer starts after the maximum number of unsuccessful attempts within the duration you configure. After the lock-out time has expired, the timers and the number of incorrect attempts resets to 60 seconds.

**httpd.cfg.lockWebUI.noOfInvalidAttempts**

5 (default) - After five failed login attempts, the user is locked out of the Web Configuration Utility.

Specify the maximum number of failed login attempts after which the user is locked out of the Web Configuration Utility.

3 - 20 seconds

**httpd.cfg.lockWebUI.noOfInvalidAttemptsDuration**

60 seconds (default) - After a user reaches the maximum failed login attempts within 60 seconds, the user is locked out of the Web Configuration Utility.

After a user reaches the maximum failed login attempts within this time duration, the user is locked out of the Web Configuration Utility. The user can try logging in again after the lock-out duration set by `httpd.cfg.lockWebUI.lockOutDuration`.

60 - 300 seconds

The timer starts again after the first incorrect password attempt.

---

**Disabling External Ports and Features**

You can disable unused external phone ports and features to increase the security of devices in your deployment.

You can disable the following ports and features:

- Web Configuration Utility
- PC port
- Aux port
- USB port
- Speakerphone
- Call forwarding
- Do Not Disturb
- Push-to-Talk (PTT)
- Auto Answer
- Applications icon

---

**Note:** At least one audio port must be enabled to send and receive calls.
• Headset
• Handset
• Host and device ports
• Bluetooth
• NFC
• Wi-Fi

Note: At least one audio port must be enabled to send and receive calls.

Disable Unused Ports and Features Parameters

Use the parameters in the following list to disable external ports or specific features.

device.net.etherModePC

1 (default) - Enable the PC port mode that sets the network speed over Ethernet.
0 - Disable the PC port mode that sets the network speed over Ethernet.

device.auxPort.enable

-1 - Disabled
0 - Auto (default)
1 - 10HD
2 - 10FD
3 - 100HD
4 - 100FD
5 - 1000FD

httpd.enabled

Base Profile = Generic
1 (default) - The web server is enabled.
0 - The web server is disabled.
Change causes system to restart or reboot.

ptt.ppttMode.enable

0 (default) - Disable push-to-talk mode.
1 - Enable push-to-talk mode.

feature.callRecording.enabled

0 (default) - Disable the phone USB port for local call recording.
1 - Enable the phone USB port for local call recording.
Change causes system to restart or reboot.
**up.handsfreeMode**

1 (default) - Enable handsfree mode.
0 - disable handsfree mode.

**feature.forward.enable**

1 (default) - Enable call forwarding.
0 - Disable call forwarding.

**feature.doNotDisturb.enable**

1 (default) - Enable Do Not Disturb (DND).
0 - Disable Do Not Disturb (DND).
Change causes system to restart or reboot.

**homeScreen.doNotDisturb.enable**

1 (default) - Enables the display of the DND icon on the phone's Home screen.
0 - Disables the display of the DND icon on the phone's Home screen.

**call.autoAnswerMenu.enable**

1 (default) - Enables the phone's Autoanswer menu.
0 - Disables the phone's Autoanswer menu.

---

**Visual Security Classification**

The participant with the lowest security classification determines the security classification for a call. For example, a Top Secret classification displays when all participants in a call have at least a Top Secret classification level.

**Note:** You can safely exchange information classified no higher than the call's security classification. For example, when a Top Secret participant calls a Restricted level participant, they can't exchange information higher than Restricted.

Participants can adjust their security classification level to a lower value during a call. The participant's classification level resets to the higher value once the call is complete.

**Visual Security Classification Parameters**

To enable the visual security classification feature, you must configure settings on the BroadSoft BroadWorks server v20 or higher and on the phones.

If a phone has multiple registered lines, administrators can assign a different security classification to each line.

An administrator can configure security classifications as names or strings, then set the priority of each classification on the server in addition to the default security classification level Unclassified. The default
security classification Unclassified displays until you set classifications on the server. When a user establishes a call to a phone not connected to this feature, the phone displays as Unclassified.

The following list includes the parameters you can use to configure visual security classification.

**voIpProt.SIP.serverFeatureControl.securityClassification**

0 (default) - The visual security classification feature for all lines on a phone is disabled.
1 - The visual security classification feature for all lines on a phone is enabled.

Change causes system to restart or reboot.

**reg.x.serverFeatureControl.securityClassification**

0 (default) - The visual security classification feature for a specific phone line is disabled.
1 - The visual security classification feature for a specific phone line is enabled.

**Encryption**

Poly supports the use of encryption to protect configuration files and phone calls.

**Encrypting Configuration Files**

Download encrypted files from the provisioning server and encrypt files before uploading them to the provisioning server.

Encrypt configuration files from the system web interface and the local device interface. Determine whether encrypted files are the same as unencrypted files and use the Polycom Software Development Kit (SDK) to facilitate key generation.

**Note:** The master configuration file, contact directory files, and configuration override files can't be encrypted.

To encrypt and decrypt configuration files on a UNIX or Linux server, generate your own 32 hex-digit key, 128-bit key, or use the Polycom SDK.

**Note:** To request the SDK and quickly install the generated key, see When Encrypting Polycom UC Software Configuration Files: Quick Tip 67442 at Polycom Engineering Advisories and Technical Notifications.

You can use the following parameters to set the key on the phone:

- `device.set`
- `device.sec.configEncryption.key`
- `device.sec.configEncryption.key.set`

If the phone doesn't have a key, download the key in plain text. To avoid security issues, use HTTPS to download the key file. Poly recommends that you name each key uniquely to help match the key with the encrypted files.
After encrypting a configuration file, it’s useful to rename the file to avoid confusing it with the original version, for example, rename `site.cfg` to `site.enc`.

**Note:** If a phone can’t decrypt a downloaded file, the phone logs the action, and an error message displays. The phone continues to do this until the provisioning server provides an encrypted file that can be read, an unencrypted file, or until the file is removed from the list in the master configuration file.

### Change the Encryption Key on the Phone and Server

To maintain secure files, you can change the encryption key on the phones and the server.

You must update the files on the server to the new key or make the files available in unencrypted format. Updating to the new key requires that you decrypt the files with the old key, then re-encrypt it with the new key.

**Procedure**

1. Place all encrypted configuration files that you want to use with the new key on the provisioning server. The phone may reboot multiple times.

2. Put the new key into a configuration file that is in the list of files downloaded by the phone, specified in `000000000000.cfg` or `<MACaddress>.cfg`.

3. Use the `device.sec.configEncryption.key` parameter to specify the new key.

4. Provision the phone again so that it downloads the new key.

**Note:** You may need to update configuration files, contact directory files, and configuration override files if they were already encrypted. You can delete configuration override files from the provisioning server so that the phone replaces them when it successfully boots.

The phone automatically reboots another time to use the new key.

### Configuration File Encryption Parameters

The following list provides the parameters you can use to encrypt your configuration files.

- **device.sec.configEncryption.key**
  
  Set the configuration encryption key used to encrypt configuration files.

  - string

  Change causes system to restart or reboot.

- **sec.encryption.upload.callLists**
  
  - 0 (default) - The call list is uploaded without encryption.
  
  - 1 - The call list is uploaded in encrypted form.

  Change causes system to restart or reboot.
sec.encryption.upload.config

0 (default) - The file is uploaded without encryption and replaces the phone specific configuration file on the provisioning server.

1 - The file is uploaded in encrypted form and replaces the existing phone specific configuration file on the provisioning server.

sec.encryption.upload.dir

0 (default) - The contact directory is uploaded without encryption and replaces the phone specific contact directory on the provisioning server.

1 - The contact directory is uploaded in encrypted form and replaces the existing phone specific contact directory on the provisioning server.

Change causes system to restart or reboot.

sec.encryption.upload.overrides

0 (default) - The MAC address configuration file is uploaded without encryption and replaces the phone specific MAC address configuration file on the provisioning server.

1 - The MAC address configuration file is uploaded in encrypted form and replaces the existing phone specific MAC address configuration file on the provisioning server.

Voice over Secure IP

You can configure phones to dynamically use either Secure Real Time Protocol (SRTP) or Real Time Protocol (RTP) depending on the media security mechanisms negotiated between phone and outbound proxy using Voice over Secure IP (VoSIP). When you enable this feature, the voice signals are transferred securely between endpoints without the need to introduce multiple lines in the Session Description Protocol (SDP).

The following are advantages for Voice over Secure IP (VoSIP):

• The voice signals are encrypted allowing a safe and secure transmission of signals between phones.

• Signaling and media to the cloud-hosted product are encrypted.

VoSIP Parameter

Use the following parameters to configure VoSIP.

reg.X.rfc3329MediaSec.enable

0 - Disables the media security mechanisms negotiated between Phone and Outbound proxy without the need of multiple m-lines in the Session Description Protocol.

1 - Enables the media security mechanisms negotiated between Phone and Outbound proxy without the need of multiple m-lines in the Session Description Protocol.
Securing Phone Calls with SRTP

Secure Real-Time Transport Protocol (SRTP) encrypts audio stream(s) to prevent interception and eavesdropping on phone calls.

You need to enable this feature to use it. When in use, phones negotiate the type of encryption and authentication to use for the session with the other endpoint.

SRTP authentication proves to the phone receiving the RTP/RTCP stream that the packets are from the expected source and have not been tampered with. Encryption modifies the data in the RTP/RTCP streams so that if the data is captured or intercepted it sounds like noise and cannot be understood. Only the intended receiver knows the key to restore the data.

If the call is completely secure (RTP authentication and encryption and RTCP authentication and RTCP encryption are enabled), a padlock symbol displays. Phone will send only one SRTP m-line for audio and video instead of multiple m-lines when VoSIP is enabled.

SRTP Parameters

Use the session parameters in the following list to enable or disable authentication and encryption for RTP and RTCP streams.

You can also turn off the session parameters to reduce the phone’s processor usage.

`mr.srtp.audio.require`

Enable or disable a requirement for SRTP encrypted audio media between MR hubs and devices.

1 (default) - Enabled
0 - Disabled

Change causes system to restart or reboot.

`mr.srtp.video.require`

Enable or disable a requirement for SRTP encrypted video media between hubs and devices.

1 (default) - Enabled
0 - Disabled

Change causes system to restart or reboot.

`sec.srtp.answerWithNewKey`

1 (default) - Provides a new key when answering a call.
0 - Does not provide a new key when answering the call.

`sec.srtp.enable`

1 (default) - The phone accepts the SRTP offers.
0 - The phone declines the SRTP offers.

The defaults for SIP 3.2.0 is 0 when Null or not defined.
Change causes system to restart or reboot.

**sec.srtp.key.lifetime**
Specifies the lifetime of the key used for the cryptographic parameter in SDP.
Null (default)
0 - The master key lifetime is not set.
Positive integer minimum 1024 or power of 2 notation - The master key lifetime is set.
Setting this parameter to a non-zero value may affect the performance of the phone.
Change causes system to restart or reboot.

**sec.srtp.mki.enabled**
0 (default) - The phone sends two encrypted attributes in the SDP, one with MKI and one without MKI when the base profile is set as Generic.
1 - The phone sends only one encrypted value.
Change causes system to restart or reboot.

**sec.srtp.mki.startSessionAtOne**
0 (default) - The phone uses MKI value of 1.
1 - The MKI value increments for each new crypto key.

**sec.srtp.offer**
0 (default) - The secure media stream is not included in SDP of an SIP invite.
1 - The phone includes secure media stream along with the non-secure media description in SDP of an SIP invite.
Change causes system to restart or reboot.

**sec.srtp.offer.HMAC_SHA1_32**
0 (default) - The AES_CM_128_HMAC_SHA1_32 crypto suite in SDP is not included.
1 - The AES_CM_128_HMAC_SHA1_32 crypto suite in SDP is included.
Change causes system to restart or reboot.

**sec.srtp.offer.HMAC_SHA1_80**
1 (default) - The AES_CM_128_HMAC_SHA1_80 crypto suite in SDP is included.
0 - The AES_CM_128_HMAC_SHA1_80 crypto suite in SDP is not included.
Change causes system to restart or reboot.

**sec.srtp.padRtpToFourByteAlignment**
0 (default) - The RTP packet padding is not required when sending or receiving video.
1 - The RTP packet padding is required when sending or receiving video.
Change causes system to restart or reboot.

**sec.srtp.require**

0 (default) - The secure media streams are not required.
1 - The phone is only allowed to use secure media streams.
Change causes system to restart or reboot.

**sec.srtp.requireMatchingTag**

1 (default) - The tag values must match in the crypto parameter.
0 - The tag values are ignored in the crypto parameter.
Change causes system to restart or reboot.

**sec.srtp.sessionParams.noAuth.offer**

0 (default) - The authentication for RTP offer is enabled.
1 - The authentication for RTP offer is disabled.
Change causes system to restart or reboot.

**sec.srtp.sessionParams.noAuth.require**

0 (default) - The RTP authentication is required.
1 - The RTP authentication is not required.
Change causes system to restart or reboot.

**sec.srtp.sessionParams.noEncrypRTCP.offer**

0 (default) - The encryption for RTCP offer is enabled.
1 - The encryption for RTCP offer is disabled.
Change causes system to restart or reboot.

**sec.srtp.sessionParams.noEncrypRTCP.require**

0 (default) - The RTCP encryption is required.
1 - The RTCP encryption is not required.
Change causes system to restart or reboot.

**sec.srtp.sessionParams.noEncrypRTP.offer**

0 (default) - The encryption for RTP offer is enabled.
1 - The encryption for RTP offer is disabled.
Change causes system to restart or reboot.

**sec.srtp.sessionParams.noEncrypRTP.require**

0 (default) - The RTP encryption is required.
1 - The RTP encryption is not required.
Change causes system to restart or reboot.

`sec.srtp.simplifiedBestEffort`
1 (default) - The SRTP is supported with Microsoft Description Protocol Version 2.0 Extensions.
0 - The SRTP is not supported with Microsoft Description Protocol Version 2.0 Extensions.

`reg.x.secureTransportRequired`
0 (Default) - The phones register based on the transport priority received in the DNS response.
1 - The phones register only on the TLS transport in the DNS response if the transport is configured as DNSNaptr.
If the transport is configured as TLSOnly, then the phone registers to the configured SIP server. The phone doesn’t register if the transport is either TCP or UDP.

Enabling Users to Lock Phones

You can enable users to lock their phones to prevent access to menus or directories.
After the phone is locked, users can only place calls to emergency and authorized numbers. You can specify which authorized numbers users can call.
If a user forgets their password, you can unlock the phone either by entering the administrator password or by disabling and re-enabling the phone lock feature. The latter method facilitates remote unlocking and avoids disclosing the administrator password to the user.

**Note:** If a locked phone has a registered shared line, calls to the shared line display on the locked phone and the phone's user can answer the call.

Phone Lock Parameters

Use the parameters in the following list to enable the phone lock feature, set authorized numbers for users to call when a phone is locked, and set scenarios when the phone should be locked.

`phoneLock.Allow.AnswerOnLock`
1 (default) - Users can answer any incoming call without needing to unlock the phone.
0 - Users must unlock the phone before answering an incoming call.

`phoneLock.authorized.x.description`
The name or description of an authorized number.
Null (default)
String
Up to five (x=1 to 5) authorized contacts that a user can call while their phone is locked. Each contact needs a description to display on the screen, and a phone number or address value for the phone to dial.
phoneLock.authorized.x.value

The number or address for an authorized contact.
Null (default)
String
Up to five (x=1 to 5) authorized contacts that a user can call while their phone is locked. Each contact needs a description to display on the screen, and a phone number or address value for the phone to dial.

phoneLock.browserEnabled

0 (default) - The microbrowser or browser is not displayed while the phone is locked.
1 - The microbrowser or browser is displayed while the phone is locked.

phoneLock.dndWhenLocked

0 (default) - The phone can receive calls while it is locked
1 - The phone enters Do-Not-Disturb mode while it is locked

phoneLock.enabled1

0 (default) - The phone lock feature is disabled
1 - The phone lock feature is enabled.

phoneLock.idleTimeout

The amount of time (in seconds) the phone can be idle before it automatically locks. If 0, automatic locking is disabled.
0 (default)
0 to 65535

phoneLock.lockState

0 (default) - The phone is unlocked.
1 - The phone is locked.
The phone stores and uploads the value each time it changes via the MAC-phone.cfg. You can set this parameter remotely using the Web Configuration Utility.

phoneLock.powerUpUnlocked

Overrides the phoneLock.lockState parameter.
0 (default) - The phone retains the value in phoneLock.lockState parameter.
1 - You can restart, reboot, or power cycle the phone to override the value for phoneLock.lockState in the MAC-phone.cfg and start the phone in an unlocked state.
You can then lock or unlock the phone locally. Poly recommends that you do not leave this parameter enabled.
Locking the Basic Settings Menu

By default, all users can access the Basic settings menu available on a Poly phone. From this menu, users can customize non-administrative features on their phone. You can choose to lock the Basic settings menu to allow certain users access to the basic settings menu.

Basic Settings Menu Lock Parameter

Use the parameter below to lock the Basic settings menu.

**up.basicSettingsPasswordEnabled**

- Specifies that a password is required or not required to access the Basic Settings menu.
- 0 (Default) - No password is required to access the Basic Settings menu.
- 1 - Password is required for access to the Basic Settings menu.

Secondary Port Link Status Report

Poly devices can detect an externally connected host connection/disconnection, informing the authenticator switch to initiate the authentication process or drop an existing authentication. This feature extends Cisco Discovery Protocol (CDP) to include a Second Port Status Type, Length, Value (TLV) that informs an authenticator switch of the status of devices connected to a device’s secondary PC port.

This feature ensures the following:

- The port authenticated by the externally attached device switches to unauthenticated upon device disconnection so that other unauthorized devices cannot use it.
- The externally attached device can move to another port in the network and start a new authentication process.
- To reduce the frequency of CDP packets, the phone does not send link-up status CDP packets before a certain time period. The phone immediately sends all link-down indication to ensure that the port security is not compromised.
- If the externally attached device (the host) supports 802.1X authentication, then the device can send an EAPOL-Logoff on behalf of the device after it is disconnected from the secondary PC port. This informs the authenticator switch to drop the authentication on the port corresponding with the previously attached device.

Secondary Port Link Status Report Parameters

You can use the parameters in the following list to configure options for the Secondary Port Link Status Report feature, including the required elapse or sleep time between two CDP UPs dispatching.

**sec.dot1x.eapollogoff.enabled**

- 0 (default) - The phone does not send an EAPOL Logoff message.
- 1 - The phone sends an EAPOL Logoff message.
Change causes system to restart or reboot.

\texttt{sec.dot1x.eapollogoff.lanlinkreset}
- 0 (default) - The phone does not reset the LAN port link.
- 1 - The phone resets the LAN port link.
Change causes system to restart or reboot.

\texttt{sec.hostmovedetect.cdp.enabled}
- 0 (default) - The phone does not send a CDP packet.
- 1 - The phone sends a CDP packet.
Change causes system to restart or reboot.

\texttt{sec.hostmovedetect.cdp.sleepTime}
- Controls the frequency between two consecutive link-up state change reports.
- 1000 (default)
- 0 to 60000
If \texttt{sec.hostmovedetect.cdp.enabled} is set to 1, there is an x microsecond time interval between two consecutive link-up state change reports, which reduces the frequency of dispatching CDP packets.
Change causes system to restart or reboot.

\section*{USB Port Parameters}
Enable or disable the USB ports on your phone.

\texttt{feature.usb.host.enabled}
\textbf{Note:} Not applicable for CCX 400.
- Use the host port for USB headset, memory sticks, mouse, and keyboards.
- 1 (default) - The USB host port is enabled.
- 0 - The USB host port is disabled.

\texttt{feature.usb.device.enabled}
\textbf{Note:} Values for CCX 500, 600, and 700.
- The device port allows you to use the phone as an audio device for your laptop.
- 1 (default) - The USB device port is enabled.
- 0 - The USB device port is disabled.
feature.usb.device.enabled

**Note:** Values for CCX 400 only.

The device port allows you to use the phone as an audio device for your laptop.
- 2 (default) - The USB device port is configured in host mode.
- 0 - The USB device port is configured in device mode.
- 1 - The USB device port is disabled.

### 802.1X Authentication

Poly phones support IEEE 802 standards.

1X authentication and the following EAP authentication methods:

- EAP-TLS (requires Device and CA certificates)
- EAP-PEAPv0/MSCHAPv2 (requires CA certificates)
- EAP-PEAPv0/GTC (requires CA certificates)
- EAP-TTLS/MSCHAPv2 (requires CA certificates)
- EAP-TTLS/GTC (requires CA certificates)
- EAP-FAST (optional Protected Access Credential (PAC) file, if not using in-band provisioning)
- EAP-MD5

#### 802.1X Authentication Parameters

To set up an EAP method that requires a device or CA certificate, you need to configure TLS Platform Profile 1 or TLS Platform Profile 2 to use with 802.1X.

You can use the parameters in the following list to configure 802.1X Authentication.


device.net.dot1x.enabled

Enable or disable 802.1X authentication

- 0 - Disabled
- 1 - Enabled

Change causes system to restart or reboot.

device.net.dot1x.identity1

Set the identity (user name) for 802.1X authentication

String

Change causes system to restart or reboot
device.net.dot1x.method
  Specify the 802.1X EAP method
  EAP-None - No authentication
  EAP-TLS,
  EAP-PEAPv0-MSCHAPv2,
  EAP-PEAPv0-GTC,
  EAP-TTLS-MSCHAPv2,
  EAP-TTLS-GTC,
  EAP-FAST,
  EAP-MD5

device.net.dot1x.password
  Set the password for 802.1X authentication. This parameter is required for all methods except
  EAP-TLS
  String
  Change causes system to restart or reboot.

device.net.dot1x.eapFastInBandProv
  Enable EAP In-Band Provisioning for EAP-FAST
  0 (default) - Disabled
  1 - Unauthenticated, active only when the EAP method is EAP-FAST

device.pacfile.data
  Specify a PAC file for EAP-FAST (optional)
  Null (default)
  0-2048 - String length

device.pacfile.password
  The optional password for the EAP-FAST PAC file.
  Null (default)
  0-255 - String length

Simple Certificate Enrollment Protocol
The Simple Certificate Enrollment Protocol (SCEP) is a protocol that enables you to automatically enroll
devices to retrieve new digital certificates or re-enroll to renew expired or expiring certificates.
Simple Certificate Enrollment Protocol Parameters

Use the following parameters to configure Simple Certificate Enrollment Protocol (SCEP).

**SCEP.CAFingerprint**

Configure the CA certificate fingerprint to confirm the authenticity of the CA response during enrollment.

- NULL (default)
- 0 - 255 characters

**SCEP.certPoll.retryCount**

Specify the number of times to poll the SCEP server when the SCEP server returns a Certificate Enrollment Response Message with pkiStatus set to 'pending'.

- 12 (default)
- 1 - 24

**SCEP.certPoll.retryInterval**

Specify the number of seconds to wait between poll attempts when the SCEP server returns a Certificate Enrollment Response Message with pkiStatus set to 'pending'.

- 300 seconds (default)
- 300 - 3600 seconds

**SCEP.certRenewalRetryInterval**

Specify the time interval to retry certificate renewal.

- 86400 seconds (default)
- 28800 - 259200 seconds

**SCEP.certRenewalThreshold**

Specify the percentage of the certificate validity interval to initiate a renewal.

- 80 (default)
- 50 - 100

**SCEP.challengePassword**

Specify the challenge password to send with the Certificate Signing Request (CSR) when requesting a certificate.

- NULL (default)
- 0 - 255 characters

**SCEP.csr.commonName**

Specify the common name to use for CSR generation.

- NULL (default)
SCEP.csr.country
Specify the country name to use for CSR generation.
NULL (default)
0 - 2

SCEP.csr.email
Specify the email address to use for CSR generation.
NULL (default)
0 - 64

SCEP.csr.organization
Specify the organization name to use for CSR generation.
NULL (default)
0 - 64

SCEP.csr.state
Specify the state name to use for CSR generation.
NULL (default)
0 - 128 characters

SCEP.enable
0 (default) - Disable the SCEP feature.
1 - Enable the SCEP feature.

SCEP.enrollment.retryCount
Specify the number of times to retry the enrolment process in case of enrolment failure.
12 (default)
1 - 24

SCEP.enrollment.retryInterval
Specify the time interval to retry the enrolment process.
300 seconds (default)
300 - 3600 seconds

SCEP.http.password
Specify the password that authenticates with the SCEP server.
NULL (default)
STRING, max 255 characters

**SCEP.http.username**

Specify the user name that authenticates with the SCEP server.

NULL (default)

STRING, max 255 characters

**SCEP.url**

Specify the URL of the SCEP server.

NULL (default)

0 - 255 characters
Certificates

Topics:

• Using the Factory-Installed Certificate
• Customizing Certificate Use
• Create a Certificate Signing Request
• Custom URL Location for LDAP Server CA Certificate
• Online Certificate Status Protocol

Use security certificates when deploying a solution to ensure the integrity and privacy of communications involving Poly devices.

Poly phones come with an authenticated, built-in device certificate. You can also choose to customize your security by requesting additional certificates from a certificate authority of your choice.

You can customize security configuration options to determine the type of device certificate used for each secure communication option. By default, all operations use the factory-installed device certificate unless you specify otherwise.

Note: You can install custom device certificates on your phones in the same way you install custom CA certificates. For more information, see Technical Bulletin 17877: Using Custom Certificates With Polycom Phones at Polycom Support.

You phone uses certificates in the following situations:

• Mutual TLS authentication - The server can verify that a device is truly a Poly device and not a malicious endpoint or software masquerading as a Poly device.

  Use this option for provisioning or SIP signaling using TLS signaling. For example, certain partner provisioning systems and Polycom Zero Touch Provisioning (ZTP) use mutual TLS.

• Secure HTTP (HTTPS) - Access to the web server on the phone at https://<IP ADDRESS OF PHONE>.

  The phone uses the web server for certain configuration and troubleshooting activities.

• Polycom applications API - Provides secure communications.

You can configure the following options for two platform device certificates and six application device certificates on the phone:

• 802.1X authentication
• Provisioning
• Syslog
• SIP signaling
• Browser communications
• Presence
• LDAP
Note: You must apply platform device certificates for syslog, 802.1X, and provisioning using TLS platform profiles, but you can’t use TLS application profiles to applied certificates for those options.

Using the Factory-Installed Certificate

Poly installs a device certificate at the manufacture that is unique to the device (based on the MAC address). Because the certificate is factory installed, it’s the easiest option for out-of-box activities, especially phone provisioning.

You can use the factory-installed certificate for all your security needs. The certificate is signed by the Poly Certificate Authority (CA), so to configure your web servers and/or clients to trust the factory-installed certificates, you must download the Poly Root CA certificate available at http://pki.polycom.com/pki. You may also need to download the Intermediate CA certificates if determined by the authenticating server.

The location of the Certificate Revocation List (CRL)—a list of all expired certificates signed by the Poly Root CA—is part of the Poly Root CA digital certificate. If you enable mutual TLS, you must have a root CA download (the Polycom Root CA certificate or your organization’s CA) on the HTTPS server.

The certificate is set to expire on March 9, 2044.

For more information on using mutual TLS with Microsoft Internet Information Services (IIS) 6.0, see Mutual Transport Layer Security Provisioning Using Microsoft Internet Information Services 6.0: Technical Bulletin 52609 at Polycom Engineering Advisories and Technical Notifications.

Check for a Device Certificate

You can check if your phone has a factory-installed certificate. The certificate and associated private key are stored on the phone in its non-volatile memory as part of the manufacturing process.

Procedure

2. Choose a credential and select Info to view the certificate.
   
   One of the following messages displays:
   
   - *Installed* or *Factory Installed* - The certificate is available in flash memory, all the certificate fields are valid, and the certificate isn’t expired.
   - *Not Installed* - The certificate isn’t available in flash memory or the flash memory location that stores the device certificate is blank.
   - *Invalid* - The certificate isn’t valid.

Note: If your phone reports the device certificate as self-signed rather than *Factory Installed*, return the equipment to receive a replacement.
Customizing Certificate Use

You can add custom certificates to the phone and set up the phone to use the certificates for different features.

For example, the phone’s factory-installed certificate can be used for authentication when phone provisioning is performed by an HTTPS server or you can use a different certificate when accessing content through a browser.

Determining TLS Platform Profiles or TLS Application Profiles

You use TLS Platform or TLS Application profiles to customize where your installed certificates are used for authentication.

After you install certificates on the phone, you can determine which TLS platform profiles or TLS application profiles use these certificates. By default, TLS Platform Profile 1 uses every CA certificate and the default device certificate. Also, each TLS application uses TLS Platform Profile 1 as the default profile. You can quickly apply a CA certificate to all TLS applications by installing it on the phone and keeping the default TLS profile and default TLS application values.

Alternatively, you can choose which TLS platform profile or application profile to use for each TLS application. You can use platform profiles for any of the following purposes: phone provisioning, for applications running on the microbrowser and browser, and for 802.1X, LDAP, and SIP authentication. You can use application profiles for all applications except 802.1X, syslog, and provisioning.

For more information on using custom certificates, see Technical Bulletin 17877: Using Custom Certificates With Polycom Phones on Polycom Support.

TLS Platform Profile and Application Profile Parameters

By default, all preinstalled profiles are associated with the default cipher suite and use trusted and widely recognized CA certificates for authentication.

The following list shows parameters for TLS Platform Profile 1. To configure TLS Platform Profile 2, use a 2 at the end of the parameter instead of a 1. For example, set device.sec.TLS.profile.caCertList2 instead of device.sec.TLS.profile.caCertList1.

You can use the parameters in the following list to configure the following TLS Profile feature options:

- Change the cipher suite, CA certificates, and device certificates for the two platform profiles and the six application profiles.
- Map profiles directly to the features that use certificates.

device.sec.TLS.customCaCert1

Specify a custom certificate.

Null (default)

String (maximum of 12288 characters)

device.sec.TLS.profile.caCertList1

Specify which CA certificates to use.

Null (default)
device.sec.TLS.profile.cipherSuite1
Specify the cipher suite.
Null (default)
String (maximum of 1024 characters)

device.sec.TLS.profile.cipherSuiteDefault1
Null (default)
0 - Use the custom cipher suite.
1 - Use the default cipher suite.

device.sec.TLS.profile.deviceCert1
Specify which device certificates to use.
Builtin (default)
Builtin, Platform1, Platform2

sec.TLS.customCaCert.x
The custom certificate for TLS Application Profile x (x= 1 to 6).
Null (default)
String

sec.TLS.customDeviceKey.x
The custom device certificate private key for TLS Application Profile x (x= 1 to 6).
Null (default)
String

sec.TLS.profile.x.caCert.application1
1 (default) - Enable a CA Certificate for TLS Application Profile 1.
0 - Disable a CA Certificate for TLS Application Profile 1.

sec.TLS.profile.x.caCert.application2
1 (default) - Enable a CA Certificate for TLS Application Profile 2.
0 - Disable a CA Certificate for TLS Application Profile 2.

sec.TLS.profile.x.caCert.application3
1 (default) - Enable a CA Certificate for TLS Application Profile 3.
0 - Disable a CA Certificate for TLS Application Profile 3.
sec.TLS.profile.x.caCert.application4
  1 (default) - Enable a CA Certificate for TLS Application Profile 4.
  0 - Disable a CA Certificate for TLS Application Profile 4.

sec.TLS.profile.x.caCert.application5
  1 (default) - Enable a CA Certificate for TLS Application Profile 5.
  0 - Disable a CA Certificate for TLS Application Profile 5.

sec.TLS.profile.x.caCert.application6
  1 (default) - Enable a CA Certificate for TLS Application Profile 6.
  0 - Disable a CA Certificate for TLS Application Profile 6.

sec.TLS.profile.x.caCert.application7
  1 (default) - Enable a CA Certificate for TLS Application Profile 7.
  0 - Disable a CA Certificate for TLS Application Profile 7.

sec.TLS.profile.x.caCert.defaultList
  Specifies the list of default CA Certificate for TLS Application Profile x (x=1 to 7).
  Null (default)
  String

sec.TLS.profile.x.caCert.platform1
  1 (default) - Enable a CA Certificate for TLS Platform Profile 1.
  0 - Disable a CA Certificate for TLS Platform Profile 1.

sec.TLS.profile.x.caCert.platform2
  1 (default) - Enable a CA Certificate for TLS Platform Profile 2.
  0 - Disable a CA Certificate for TLS Platform Profile 2.

sec.TLS.profile.x.cipherSuite
  Specifies the cipher suite for TLS Application Profile x (x=1 to 8).
  Null (default)
  String

sec.TLS.profile.x.cipherSuiteDefault
  1 (default) - Use the default cipher suite for TLS Application Profile x (x=1 to 8).
  0 - Use the custom cipher suite for TLS Application Profile x (x=1 to 8).
sec.TLS.profile.x.deviceCert

Specifies the device certificate to use for TLS Application Profile x (x = 1 to 7).

Polycom (default)
Platform1, Platform2, Application1, Application2, Application3, Application4, Application5, Application6, Application7

TLS Protocol Configuration for Supported Applications

You can configure the TLS Protocol for the following supported applications:

- LDAP
- SIP
- SOPI
- Web server
- XMPP
- Exchange services
- Syslog
- Provisioning
- 802.1x

TLS Protocol Parameters

The following list includes the parameters for the TLS protocol supported applications.

device.sec.TLS.protocol.dot1x

Configures the lowest TLS/SSL version to use for handshake negotiation between phone and 802.1x authentication. The phone handshake starts with the highest TLS version irrespective of the value you configure.

TLSv1_0 (default)
SSLv2v3
TLSv1_1
TLSv1_2

device.sec.TLS.protocol.prov

Configures the lowest TLS/SSL version to use for handshake negotiation between phone and provisioning. The phone handshake starts with the highest TLS version irrespective of the value you configure.

TLSv1_0 (default)
SSLv2v3
TLSv1_1
TLSv1_2
**device.sec.TLS.protocol.syslog**

Configures the lowest TLS/SSL version to use for handshake negotiation between phone and Syslog. The phone handshake starts with the highest TLS version irrespective of the value you configure.

- TLSv1_0 (default)
- SSLv2v3
- TLSv1_1
- TLSv1_2

**sec.TLS.protocol.browser**

Configure the lowest TLS/SSL version to use for handshake negotiation between the phone and phone browser. The phone handshake starts with the highest TLS version irrespective of the value you configure.

- TLSv1_0 (default)
- SSLv2v3
- TLSv1_1
- TLSv1_2

The microbrowser restarts when there is a change in the browser TLS protocol or TLS cipher settings, and the last web page displayed is not restored.

**sec.TLS.protocol.exchangeServices**

Configures the lowest TLS/SSL version to use for handshake negotiation between phone and Exchange services. The phone handshake starts with the highest TLS version irrespective of the value you configure.

- TLSv1_0 (default)
- SSLv2v3
- TLSv1_1
- TLSv1_2

**sec.TLS.protocol.ldap**

Configure the lowest TLS/SSL version to use for handshake negotiation between phone and Lightweight Directory Access Protocol (LDAP). The phone handshake starts with the highest TLS version irrespective of the value you configure.

- TLSv1_0 (default)
- SSLv2v3
- TLSv1_1
- TLSv1_2

**sec.TLS.protocol.sip**

Configures the lowest TLS/SSL version to use for handshake negotiation between the phone and SIP signaling. The phone handshake starts with the highest TLS version irrespective of the value you configure.
sec.TLS.protocol.sopi

Configures the lowest TLS/SSL version to use for handshake negotiation between phone and SOPI. The phone handshake starts with the highest TLS version irrespective of the value you configure.

- TLSv1_0 (default)
- SSLv2v3
- TLSv1_1
- TLSv1_2

sec.TLS.protocol.webServer

Configures the lowest TLS/SSL version to use for handshake negotiation between phone and web server. The phone handshake starts with the highest TLS version irrespective of the value you configure.

- TLSv1_0 (default)
- SSLv2v3
- TLSv1_1
- TLSv1_2

sec.TLS.protocol.xmpp

Configures the lowest TLS/SSL version to use for handshake negotiation between phone and XMPP. The phone handshake starts with the highest TLS version irrespective of the value you configure.

- TLSv1_0 (default)
- SSLv2v3
- TLSv1_1
- TLSv1_2

**TLS Parameters**

The next list includes configurable TLS parameters.

For the list of configurable ciphers, refer to the Secure Real-Time Transport Protocol table.

**sec.TLS.browser.cipherList**

The cipher list is for browser. The format for the cipher list uses OpenSSL syntax found at: [https://www.openssl.org/docs/man1.0.2/apps/ciphers.html](https://www.openssl.org/docs/man1.0.2/apps/ciphers.html).

- NoCipher (default)
sec.TLS.customDeviceCert.x
The custom device certificate for TLS Application Profile x (x= 1 to 6).
Null (default)

sec.TLS.LDAP.cipherList
The cipher list for the corporate directory. The format for the cipher list uses OpenSSL syntax found here: https://www.openssl.org/docs/man1.0.2/apps/ciphers.html.
NoCipher (default)

sec.TLS.profileSelection.SOPI
Select the platform profile required for the phone.
PlatformProfile1 (default)
1 - 7

sec.TLS.profile.webServer.cipherSuiteDefault
1 (default) - The phone uses the default cipher suite for web server profile.
0 - The custom cipher suite is used for web server profile.

sec.TLS.prov.cipherList
The cipher list for provisioning. The format for the cipher list uses OpenSSL syntax found here: https://www.openssl.org/docs/man1.0.2/apps/ciphers.html.
NoCipher (default)

sec.TLS.SIP.cipherList
The cipher list for SIP. The format for the cipher list uses OpenSSL syntax found here: https://www.openssl.org/docs/man1.0.2/apps/ciphers.html.
NoCipher (default)

sec.TLS.SIP.strictCertCommonNameValidation
1 (default) - The common name validation is enabled for SIP.
0 - The common name validation is not enabled for SIP.

sec.TLS.SOPI.cipherList
Selects a cipher key from the list of available ciphers.
NoCipher (default)
1 - 1024 character string

**sec.TLS.SOPI.strictCertCommonNameValidation**
Controls the strict common name validation for the URL provided by the server.
1 (default) - The SOPI verifies the server certificate to match commonName/SubjectAltName against the server hostname.
0 - The SOPI will not verify the server certificate for commonName/SubjectAltName against the server hostname.

**sec.TLS.syslog.cipherList**
The cipher list for syslog. The format for the cipher list uses OpenSSL syntax found here: [https://www.openssl.org/docs/man1.0.2/apps/ciphers.html](https://www.openssl.org/docs/man1.0.2/apps/ciphers.html)
NoCipher (default)
String

**TLS Profile Selection Parameters**
You can configure the parameters listed below to choose the platform profile or application profile to use for each TLS application.

**sec.TLS.profileSelection.browser**
Specifies to select a TLS platform profile or TLS application profile for the browser or a microbrowser.
PlatformProfile1 (default)
- PlatformProfile1
- PlatformProfile2
- ApplicationProfile1
- ApplicationProfile2
- ApplicationProfile3
- ApplicationProfile4
- ApplicationProfile5
- ApplicationProfile6
- ApplicationProfile7

**sec.TLS.profileSelection.LDAP**
Specifies to select a TLS platform profile or TLS application profile for the corporate directory.
PlatformProfile1 (default)
- PlatformProfile1
- PlatformProfile2
- ApplicationProfile1
sec.TLS.profileSelection.SIP
Specifies to select a TLS platform profile or TLS application profile for SIP operations.
PlatformProfile1 (default)
  • PlatformProfile1
  • PlatformProfile2
  • ApplicationProfile1
  • ApplicationProfile2
  • ApplicationProfile3
  • ApplicationProfile4
  • ApplicationProfile5
  • ApplicationProfile6
  • ApplicationProfile7

sec.TLS.profileSelection.syslog
Specifies to select a TLS platform profile for the syslog operations.
PlatformProfile1 (default)
  PlatformProfile1 or PlatformProfile2

Configurable TLS Cipher Suites
You can configure which cipher suites to offer and accept during TLS session negotiation. The following table lists supported cipher suites. NULL cipher is a special case that does not encrypt the signaling traffic.

<table>
<thead>
<tr>
<th>Cipher</th>
<th>Cipher Suite</th>
</tr>
</thead>
<tbody>
<tr>
<td>AES128</td>
<td>AES128-SHA</td>
</tr>
<tr>
<td>AES256</td>
<td>AES256-SHA</td>
</tr>
<tr>
<td>ECDH</td>
<td>ECDH_RSA_WITH_AES_256_GCM_SHA384, ECDH_ECDSA_WITH_AES_256_GCM_SHA384,</td>
</tr>
<tr>
<td></td>
<td>ECDH_RSA_WITH_AES_256_CBC_SHA384</td>
</tr>
</tbody>
</table>
### TLS Cipher Suite Parameters

You can use the parameters listed below to configure TLS Cipher Suites.

**sec.TLS.cipherList**

- String (1 - 1024 characters)
- RC4:@STRENGTH (default)
- ALL:!aNULL:!eNULL:!DSS:!SEED
- !ECDSA:!IDEA:!MEDIUM:!LOW:
- EXP:!DH:AESCAH:!PSK:!SRP:!MD5:
- RC4:@STRENGTH

The global cipher list parameter. The format for the cipher list uses OpenSSL syntax found at: [https://www.openssl.org/docs/man1.0.2/apps/ciphers.html](https://www.openssl.org/docs/man1.0.2/apps/ciphers.html).

**sec.TLS.<application>.cipherList**

Specify the cipher list for a specific TLS Platform Profile or TLS Application Profile.

### Custom Wi-Fi Certificates

You can install custom wireless network certificates for added security.

For wireless network certificates:
• The phone shared Platform CA and Application CA certificates between Wi-Fi and Ethernet settings.
  The phone can't connect to both Ethernet and Wi-Fi at the same time.
• The phone retains installed and saved certificates until you choose to forget the network.
• CCXphones don't support certificates obtained via SCEP.

Install and Choose a Root CA Wi-Fi Certificate
Install a custom certificate for connecting to your wireless network.

Note: Client certificates and key must be in PKCS#8 PEM format.

Note: Only CA 1 and 2 and Platform 1 and 2 are valid for Wi-Fi.

If you set `device.wifi.wpa2Ent.caCert.name` to `none`, the phone user must choose the certificate when they connect to a wireless network.

Procedure
1. Install the certificates.
   ```
   device.sec.TLS.customCaCert1="<value>
   device.sec.TLS.customCaCert2="<value>
   ```
2. Choose the certificate to use.
   ```
   device.wifi.wpa2Ent.caCert.name="<Platform 1 or Platform 2>"
   ```

Install and Choose a Client Wi-Fi Certificate
For added wireless network security, install a client certificate.

Note: Client certificates and key must be in PKCS#8 PEM format.

Note: Only CA 1 and 2 and Platform 1 and 2 are valid for Wi-Fi.

If you set `device.wifi.wpa2Ent.clientCert.name` to `none`, the phone user must choose the certificate when they connect to a wireless network.

Procedure
1. Install the certificates.
   ```
   sec.TLS.customDeviceCert1="<value>
   sec.TLS.customDeviceCert2="<value>
   ```
2. Choose the certificate to use.
   ```
   device.wifi.wpa2Ent.clientCert.name ="<Platform 1 or Platform 2>"
   ```
Create a Certificate Signing Request

You can generate a certificate signing request directly from your device.

You must have a provisioning server in place before generating the certificate signing request.

By default, the phone requests a 2048-bit certificate with `sha256WithRSAEncryption` as the signature algorithm. You can use OpenSSL or another certificate signing request utility if you require a stronger certificate.

Poly phones support Subject Alternative Names (SAN) with TLS security certificates but doesn’t support asterisks (*) or wildcard characters in the Common Name field of a Certificate Authority’s public certificate. If you want to enter multiple hostnames or IP addresses on the same certificate, use the SAN field.

**Procedure**

1. Go to **Settings > Advanced > Admin Settings > Generate CSR**.
2. When prompted, enter the administrative password and press **Enter**.
3. Enter the following information: Common Name field - the Organization, Email Address, Country, and State fields are optional.
4. Select **Generate**.
   
   A **CSR generation completed** message displays. The **MAC.csr** (certificate request) and **MAC-private.pem** (private key) files upload to the phone’s provisioning server.
5. Forward the CSR to a Certificate Authority (CA) to create a certificate.

   If your organization doesn’t have its own CA, you must forward the CSR to a security company like Symantec.

Download Certificates

You can download and install up to eight CA certificates and eight device certificates onto a Poly phone.

After installing the certificates, you can refresh the certificates when they expire or are revoked, and you can delete any CA certificate or device certificate that you install.

You can download certificate(s) to a phone in the following ways:

- Using a configuration file
- Through the phone’s local interface
- Through the system web interface

**Procedure**

1. Go to **Settings > Advanced > Administrative Settings > TLS Security and select Custom CA Certificates or Custom Device Certificates**.
2. Select **Install**.
3. Enter the URL where the certificate is stored.

   For example, `http://bootserver1.polycom.com/ca.crt`.

   The certificate downloads, and the certificate’s MD5 fingerprint displays to verify that you are installing the correct certificate.
4. Select **Accept**.

   The certificate installs successfully.
Custom URL Location for LDAP Server CA Certificate

You can set a custom location on the phones to download a CA certificate or a chain of CA certificates required to authenticate the LDAP server.

By default, all Poly-installed profiles are associated with the default cipher suite and use trusted and widely recognized CA certificates for authentication. You can download and install up to seven custom CA certificates onto a phone. The certificates install in descending order starting with the highest Application CA slot (up to 7) and continues to Application CA 1 slot.

**Note:** If the custom application CA certificate slots already have CA certificates installed on your phones, downloading LDAP server CA certificates overwrites any existing certificates on the phone.

---

Custom URL Location for LDAP Server Certificates Parameter

Use the parameter below to configure a custom URL location for LDAP server certificates.

In addition to the parameter below, you must also configure the following Corporate Directory parameters:

- `sec.TLS.profilenameSelection.LDAP = ApplicationProfile1`

`sec.TLS.LDAP.customCaCertUrl`

Enter the URL location from where the phone can download LDAP server certificates.

- String (default)
- 0 - Minimum
- 255 - Maximum

You must configure parameters `dir.corp.address` and `feature.corporateDirectory.enabled` as well to enable this parameter.

---

Confirm the Installed LDAP Server Certificates on the Phone

After you configure the custom URL location for the LDAP server certificates, confirm that the phone downloaded and installed the correct certificates.

**Procedure**

1. On the phone’s local interface, go to **Settings > Advanced**.
2. Enter the administrator password.
3. Go to **Administrative Settings > TLS Security > Custom CA Certificates > Application CA placeholders**.
4. Confirm that phone downloaded and installed the correct certificates.
Online Certificate Status Protocol

The Online Certificate Status Protocol (OCSP) is used to authenticate the revocation status of an X.509 digital certificate. When a user sends a request to a server, the OCSP retrieves the information whether the certificate is valid or revoked.

Online Certificate Status Protocol Parameter

OCSP is a more advanced protocol than the existing CRL. OCSP further offers a grace period for an expired certificate to access servers for a limited time before certificate renewal. OCSP is disabled by default.

device.sec.TLS.OCSP.enabled

0 (default) OCSP is disabled.
1 – OCSP is enabled

Change causes system to restart or reboot.

Ensure that device.set="1", and device.sec.TLS.OCSP.enabled.set="1" to enable OCSP.
Upgrading the Software

Topics:

• Upgrading the Software on a Single Phone
• User-Controlled Software Update
• Upgrade UC Software Using a USB Flash Drive

You can upgrade software with the user-controlled software upgrade feature. The new software versions may offer only small enhancements to improve the user experience, or they may be large software upgrades that offer new features.

The upgrade process varies depending on the software version that is currently running on your phones and the version that you want to upgrade to.

Upgrading the Software on a Single Phone

You can use the Software Upgrade tool in the system web interface (Web Configuration Utility) to update the software version running on a single phone.

For instructions, see Use the Software Upgrade Tool in the Web Configuration Utility: Feature Profile 67993 at Polycom Engineering Advisories and Technical Notifications.

Configuration changes made to individual phones using the system web interface override configuration settings made using central provisioning.

User-Controlled Software Update

This feature enables phone users to choose when to accept software updates you send to the phones.

The software you send to your users' phones can be earlier or later versions. User-controlled updates apply to configuration changes and software updates you make on the server and the system web interface (Web Configuration Utility).

If a user postpones a software update, configuration changes and software version updates from both the server and the system web interface are postponed. When the user chooses to update, configuration and software version changes from both the server and system web interface are sent to the phone.

This feature doesn’t work if you enable ZTP.

User-Controlled Software Update Parameters

You can set a polling policy and polling time period at which the phone polls the server for software updates and displays a notification on the phone to update software.

For example, if you set the polling policy to poll every four hours, the phone polls the server for new software every four hours and displays a notification that says a software update is available. Users can choose to update the software right then, or they can postpone it a maximum of three times for up to six hours. The phone automatically updates the software after three postponements or after six hours, whichever comes first.
The polling policy is disabled after the phone displays the software update notification. After the software postponement ends, the phone displays the software update notification again.

**prov.usercontrol.enabled**

- 0 (default) - The phone doesn't display the software update notification and options and the phone reboots automatically to update the software.
- 1 - The phone displays the software update notification and options and the user can control the software download.

**prov.usercontrol.postponeTime**

Sets the time interval for software update notification using the HH:MM format.

- 02:00 (default)
- 00:15
- 01:00
- 02:00
- 04:00
- 06:00

---

**Upgrade UC Software Using a USB Flash Drive**

You can use a USB flash drive to upgrade the software on your CCX business media phone. Changes you make using a USB flash drive override the settings you configure using a centralized provisioning server (if applicable).

**Procedure**

1. Do one of the following:
   - Format a blank USB 2.0 USB flash drive using FAT32.
   - Delete all files from a previously formatted USB flash drive.
2. Download the UC software from Polycom Support.
3. Copy the configuration files you want to use to the root of the USB flash drive.
   
   You must copy the minimum required configuration files to the drive:
   - Master configuration file: 000000000000.cfg.
   - The *.sip.ld file for your phone.
4. Insert the USB flash drive into the USB port.
5. Enter the administrator password.

   The system detects the flash drive and starts the update within 30 seconds. The mute keys indicator lights begin to flash, indicating that the update has started.

   The system reboots several times during the update. The update is complete when the indicator lights stop flashing and the Home screen displays.
Hardware and Accessories

Topics:

- Powering Phones Using Power over Ethernet
- Connecting a Computer Through the Phone
- Power-Saving
- USB Port Lock
- Headset Support

This section provides information on configuring hardware pairing options, and supported accessories.

Powering Phones Using Power over Ethernet

Power your phone using a Power over Ethernet (PoE) switch or a power supply.

**Important:** If you are using a power supply, ensure you use the correct power supply for your phone.

Poly recommends powering your phones with PoE. If your Ethernet port doesn't support PoE, use an optional power supply.

Connecting a Computer Through the Phone

Poly phones allow you to connect your computer to your network through the phone.

The phone includes an Ethernet and LAN port with an internal Ethernet switch. This switch allows you to use your phone as an Ethernet hub.

If you're using a VLAN, set the 802.1p priorities for both default and RTP packet types to 2 or greater. Setting this priority ensures that audio packets from the phone have priority over packets from the PC port.

Power-Saving

The Power-Saving feature automatically turns off the phone's LCD display when not in use.

You can configure the following power-saving options:

- Turn on the phone's power-saving feature during non-working hours and working hours.
  
  If you want to turn on power-saving during non-working hours, you can configure the power-saving feature around your work schedule.

When you enable power-saving mode and the phone is in low power state, the red LED indicator flashes at three second intervals to show that the phone still has power.
Power-Saving Parameters

Use the following parameters to configure power-saving features and feature options.

**powerSaving.Enable**

1 (default) - Enable the LCD power-saving feature.
0 - Disable the LCD power-saving feature.

**powerSaving.idleTimeout.offHours**

The number of idle minutes during off hours after which the phone enters power saving.
1 (default)
1 - 10

**powerSaving.idleTimeout.officeHours**

The number of idle minutes during office hours after which the phone enters power saving.
30 (default)
1 - 600

**powerSaving.idleTimeout.userInputExtension**

The number of minutes after the phone is last used after which the phone enters power saving.
10 (default)
1 - 20

**powerSaving.officeHours.duration.x**

Append the day of the week for x. For example, `powerSaving.officeHours.duration.Monday`.

Set the duration of the office working hours by weekday.
Monday - Friday = 12 (default)
Saturday - Sunday = 0
0 - 24

**powerSaving.officeHours.startHour.x**

Specify the starting hour for the day's office working hours.
7 (default)
0 - 23

Set x to Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday (see `powerSaving.officeHours.duration` for an example).
USB Port Lock

The USB port lock down feature enables you to choose which of the phone's USB ports to power on or off.

The phone ports support various USB devices such as USB mass storage devices and a USB headset. The following features are not available when you disable a USB port:

- Call recording
- Picture frame
- USB headset
- Sending video from a CCX 600 with the optional Poly EagleEye Mini USB camera

USB Port Lock Parameters

Insert a short description

Note: Two parameters `feature.usbTop.power.enabled` and `feature.usbRear.power.enabled` replace `feature.usb.power.enabled`. You must replace `feature.usb.power.enabled` with these two new parameters in your configuration file and set both parameters to 0 to disable USB ports.

`feature.usbTop.power.enabled`
1 (default) - Enable power to the USB port (port 1).
0 - Disable power to the USB port and the phone does not detect USB devices connected to the USB port.

Change causes system to restart or reboot.

`feature.usbRear.power.enabled`
1 (default) - Enable power to the rear USB port (port 2).
0 - Disable power to the rear USB port and the phone does not detect USB devices connected to the rear USB port.

Change causes system to restart or reboot.

`video.enable`

To ensure the USB port is disabled when you set `feature.usbTop.power.enabled` to 0, you must also disable this parameter.

1 (default) - Enables video in outgoing and incoming calls.
0 - Disables video.
Headset Support
The phone supports the following headsets and headset connection types.

Headset and Speakerphone Parameters
You can use the parameters in the following list to enable and disable the headset or speakerphone and control other options for the headset and speakerphone.

`up.audioMode`
- Specify whether you want to use the handset or headset for audio.
  - 0 (Default) - Use the handset for audio.
  - 1 - Enabled - Use the headset for audio.
Diagnostics and Status

Topics:

• View the Phone’s Status
• Test Phone Hardware
• Upload a Phone’s Configuration Files
• Perform Network Diagnostics
• Reboot the Phone
• Restart the Phone
• Reset the Phone and Configuration
• Monitoring the Phone’s Memory Usage
• Remote Packet Capture
• Phone Boot Status
• Retrieve Logs Using the Web Configuration Utility
• Retrieve Logs from the Support Information Package

There are a variety of screens and logs that display on Poly devices that enable you to review performance information about the phone, help you diagnose and troubleshoot problems, view error messages, and test the phone’s hardware.

Review the latest Release Notes for your product at Voice Support for known problems and possible workarounds. If you don’t find your problem in this section or in the latest Release Notes, contact your certified reseller for support.

View the Phone’s Status

You can troubleshoot phone issues by viewing the phone’s Status menu.

Procedure

1. Go to Settings > Status > Select.
2. Scroll to a Status menu item and press Select.
3. View the following information:
<table>
<thead>
<tr>
<th>Menu Item</th>
<th>Available Information</th>
</tr>
</thead>
</table>
| Platform  | • Phone’s serial number or MAC address  
|           | • Current IP address  
|           | • Updater version  
|           | • Application version  
|           | • Name of the configuration files in use  
|           | • Address of the provisioning server |
| Network   | • TCP/IP Setting  
|           | • Ethernet port speed  
|           | • Connectivity status of the PC port (if it exists)  
|           | • Statistics on packets sent and received since last boot  
|           | • Last time the phone rebooted  
|           | • Call Statistics showing packets sent and received on the last call |
| Lines     | • Detailed status of each of the phone’s configured lines |
| Diagnostics | • Hardware tests to verify correct operation of the microphone, speaker, handset, and third party headset, if present  
|           | • Hardware tests to verify correct operation of the microphones and speaker  
|           | • Tests to verify proper functioning of the phone keys  
|           | • List of the functions assigned to each of the phone keys  
|           | • Real-time graphs for CPU, network, and memory use |

**Test Phone Hardware**

You can test the phone’s hardware directly from the user interface.

**Procedure**

1. Go to **Settings > Status > Diagnostics > Test Hardware**.
2. Choose from these tests:
   • **Audio Diagnostics**  Test the speaker, microphone, handset, and a third party headset.
   • **Keypad Diagnostics**  Verify the function assigned to each keypad key.
   • **Display Diagnostics**  Test the LCD for faulty pixels.
   • **Touch Screen Diagnostics**  Test the touch screen response.
   • **Brightness Diagnostics**  Test the screen brightness.
Upload a Phone’s Configuration Files

You can upload the phone’s current configuration files from the local interface or the system web interface to the provisioning server to help debug configuration problems.

You can upload a configuration file for every active source as well as the current non-default configuration set.

**Procedure**

1. Go to **Settings > Advanced > Admin Settings > Upload Configuration**.
2. Choose the files to upload:
   - All Sources
   - Configuration Files
   - Local
   - MR
   - Web
   - SIP

   For example, if you select **All Sources**, the phone uploads the `<MACaddress>-update-all.cfg` file.

   If you use the system web interface, you can also upload **Device Settings**.
3. Select **Upload**.
   The phone uploads the configuration file to the location you specified in the `prov.configUploadPath` parameter.

Perform Network Diagnostics

You can use ping and trace route to troubleshoot network connectivity problems.

**Procedure**

1. Go to **Settings > Status > Diagnostics > Network**.
2. Enter a URL or IP address.
3. Press **Enter**.

Reboot the Phone

You can reboot the phone from the phone menu when you want to send configuration changes requiring a reboot or restart to the phone.

Parameters that require a reboot or restart are marked in the parameter lists in this guide. If a configuration change does not require a reboot or restart, you can update configuration.

**Procedure**

» On the phone, go to **Settings > Advanced > Reboot Phone**.
Restart the Phone

You can restart the phone from the phone menu when you want to send configuration changes requiring a reboot or restart to the phone.

Parameters that require a reboot or restart are marked in the parameter lists in this guide. For configuration changes that do not require a reboot or restart, you can update configuration.

Procedure

» On the phone, go to Settings > Basic > Update Configuration.

If new Updater or Polycom UC Software is available on the provisioning server, the phone downloads the software. If new software is available on the provisioning server, the phone downloads the software and restarts.

Update Configuration from the Phone Menu

You can update the phone configuration from the phone menu when you want to send configuration changes to the phone.

Some configuration changes require a reboot or restart and parameters that require a reboot or restart are marked in the parameter lists in this guide. If there are configuration file changes or new software available on the provisioning server, your phone restarts or reboots if required.

Procedure

» On the phone, go to Settings > Basic > Update Configuration.

Reset the Phone and Configuration

You can reset the phone and phone configuration partially or completely.

Procedure

1. On the phone's local interface, go to Settings > Advanced > Administration Settings.
2. Select Reset to Defaults and choose a reset option:
   • Reset Local Configuration
   • Reset Web Configuration
   • Reset Device Settings
   • Format File System
   • Reset to Factory

Reset to Factory Parameter

By default, only administrators can initiate a factory reset. However, you can make the Reset to Factory setting available to users.

up.basicSettings.factoryResetEnabled

0 (default) - Doesn't display the Reset to Factory option under Basic settings.
1 - Displays the **Reset to Factory** option under **Basic** settings.

`feature.restrictPerDataUploadMenu.enabled`

1 (default) - Displays the **Restrict Personal Data Upload** menu under **Basic** settings.
0 - Doesn’t display the **Reset to Factory** menu under **Basic** settings.

`feature.clearPerInfoMenu.enabled`

1 (default) - Displays the **Clear Personal Information** menu under **Basic** settings.
0 - Doesn’t display the **Clear Personal Information** menu under **Basic** settings.

**Monitoring the Phone’s Memory Usage**

If you are using a range of phone features, customized configurations, or advanced features, you might need to manage phone memory resources.

If your deployment includes a combination of phone models, consider configuring each phone model separately with its own features instead of applying all features to all models.

For best performance, the phone should use no more 95% of its available memory. When the phone memory resources are low, you may notice one or more of the following symptoms:

- The phone reboots or freezes up.
- The phone doesn’t download all ringtones, directory entries, backgrounds, or XML dictionary files.

**Check Memory Usage from the Phone**

You can view a graphical representation of the phone’s memory usage on the phone’s local interface.

Load and configure the features and files you want to make available on the phone’s local interface.

**Procedure**

1. Go to **Settings > Status > Diagnostics**.
2. Select **Graphs > Memory Usage**.

**View Memory Usage Errors in the Application Log**

Each time the phone's minimum free memory goes below about 5%, the phone displays a message in the application log that the minimum free memory has been reached.

The application log file is enabled by default. The file is uploaded to the provisioning server directory on a configurable schedule. You can also upload a log file manually.
Phone Memory Resources

If you need to free up memory on your phone, review the following table for the amount of memory each customizable feature uses. You can then reduce the amount of memory you need the feature to use.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Typical Memory Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Custom idle display image</td>
<td>15 KB</td>
<td>The average size of the display image is 15 KB. Custom idle display image files should also be no more than 15 KB.</td>
</tr>
</tbody>
</table>
| Local Contact directory  | 42.5 KB             | The phones are optimized to display a maximum of 250 contacts. Each contact has four attributes and requires 170 B. A local contact directory of this size requires 42.5 KB. To reduce memory resources used by the local contact directory:  
  - Reduce the number of contacts in the directory.
  - Reduce the number of attributes per contact. |
| Corporate directory      | Varies by server    | The phones are optimized to corporate directory entries with five to eight contact attributes each. The size of each entry and the number of entries in the corporate directory vary by server. If the phone can't display directory search results with more than five attributes, make additional memory resources available by reducing memory requirements of another feature. |
| Ringtones                | 16 KB               | The ringtone files range in size from 30 KB to 125 KB. If you use custom ringtones, limit the file size to 16 KB. To reduce memory resources required for ringtones, reduce the number of available ringtones. |
| Background images        | 8 KB to 32 KB       | The phones are optimized to display background images of 50 KB. To reduce memory resources required for background images, reduce the number and size of available background images. |
| Local interface language | 90 KB to 115 KB, depending on language | The language dictionary file used for the phone's user interface ranges from 90 KB to 115 KB for languages that use an expanded character set. To conserve memory resources, use XML language files for only the languages you need. |
| System web interface     | 250 KB to 370 KB    |                                                                 |
Phone Memory Alert
You can configure a threshold as a percentage of the phone’s free memory. If the phone’s free memory falls below this threshold, for example, 20%, the phone displays a warning message. You can also configure the interval, in minutes, that the phone’s free memory is checked.

Phone Memory Alert Parameters
The following parameters configure the phone memory alert feature.

up.sysFreeMemThresholdPercent
Set the threshold of free memory, in percentage, below which the phone displays a warning message.
20 percent (default)
20 - 30 percent

up.lowSysMemWarn.timeInMins
Set the interval, in minutes, that the phone’s free memory is checked.
0 (default)
0 - 1440 minutes

Remote Packet Capture
You can configure phones to capture packets. Using parameters you can enable the remote packet capture feature.

Remote Packet Capture Parameters
Use these parameters to enable and set up the remote packet capture feature.

diags.dumpcore.enabled
Determine whether the phone generates a core file if it crashes.
1 (default) - The phone generates a core file.
0 - The phone doesn’t generate a core file.
Change causes system to restart or reboot.

diags.pcap.enabled
Enable or disable all on-board packet capture features.
0 (default) - Disable on-board packet capture features.
1 - Enable on-board packet capture features.

diags.pcap.remote.enabled
Enable or disable the remote packet capture server.
0 (default) - Disable the remote packet capture server.
1 - Enable the remote packet capture server.

**diags.pcap.remote.password**
Enter the remote packet capture password.

<MAC Address>(default)

alphanumeric value

**diags.pcap.remote.port**
Specify the TLS profile to use for each application.
2002 (default)
Valid TCP Port

**Phone Boot Status**
This feature displays the phone’s status pop-up information for IP Address, VLAN ID, Provisioning and SNTP status upon every reboot/restart. This feature is enabled by default.

**Note:** Restart status may not be in sync or as expected due to limitation on network activity.

**Phone Boot Status Parameters**
Use the following parameters to configure the phone boot status popup message.

**up.phoneBootStatusPopupEnabled**
1 (default) - The phone displays a popup message with phone status details after a restart or reboot.
0 - The phone does not display a popup message after a restart or reboot.

**Retrieve Logs Using the Web Configuration Utility**
You can view and export log files using a phone’s Web Configuration Utility.

**Procedure**
1. Log in to the Web Configuration Utility as an Admin.
2. Go to Diagnostics > View & Download Logs > Audit.
Retrieve Logs from the Support Information Package

You can export **Support information Package** (.tar file) using the Web Configuration Utility.

The support information package includes the following log files:

- pbu file
- app log file
- boot log file
- audit log file

**Procedure**

1. Log in to the Web Configuration Utility as an Administrator.
2. Go to **Diagnostics > Download Support Information Package**.
3. Unzip the .tar file to view the log files.
Analytics Support for Poly Cloud Services

Topics:

- Busy Lamp Field
- Shared Call Appearance
- User Interface Analytics
- UPtime Analytics
- Hardware Analytics
- Device Details Sent to the Cloud
- Device Diagnostics Details
- Device Analytics Parameters
- Cloud Service Commands

You can configure phones to accept commands from the cloud analytics service to perform specified operations on the device and retrieve device details.

Poly phones send the following details to the cloud:

- Device Asset
- Device Network
- Device Diagnostics

Poly phones send the device details to the cloud when the following occurs:

- Phone restarts or reboots
- On-demand request from the cloud
- Device details are updated or changed

Importing and Exporting Configurations

When you enable Device Analytics and set the `da.supported.services` value to `all` or `config`, you can configure the following device options:

- Download a configuration file to a phone from the cloud
- Upload the configuration of a phone to cloud

Poly phones don’t support analytics when you configure the phones with IPv6 or dual stack Ethernet configuration.

Note: For more information on Device Analytics, refer to the Polycom Device Analytics Service Guide on the Polycom Documentation Library.
Busy Lamp Field

When you enable Device Analytics and set `da.supported.services` to `all` or `blf`, the following details are sent to the cloud:

- The total number of configured Busy Lamp Field (BLF) lines.
- The total number of dropped BLF line notification.
- The total number of actions/pickup on BLF line.
- The phone increments the BLF’s line notification for every new notification for each BLF configured line.

Shared Call Appearance

When you enable Device Analytics and set `da.supported.services` to `all` or `sca`, the following details are sent to the cloud:

- The total number of registered Shared Call Appearance (SCA) lines.
- The total number of action or resume/barge-in on SCA line.
- The phone increments the SCA line notification for every new notifications of call-info, line-seize, and dialog for each SCA configured line.

User Interface Analytics

User Interface analytics enables you to upload phone activity to the cloud when you set `da.supported.services` to `all` or `uianalytics`.

User Interface analytics includes the following two metrics:

- Key-press analytics
- Feature access analytics

Key-Press Analytics

Key-press analytics enables you to track and maintain hard and soft key press count on the phone for each key.

You can upload key-press counts at intervals you configure. Counters per key are reset after each upload. You cannot record the sequence of the key presses on the phone.

Feature Access Analytics

Feature access analytics enables you to track and maintain features that users access on the phone.

When a user accesses a feature, the corresponding feature counter is incremented. You can upload feature counts at an interval you configure. Feature counters are reset after each upload.
UPtime Analytics

Phone keeps track of various services and uploads the active status to cloud periodically when `da.supported.services` value is set as `all` or `uptimeanalytics`.

The following services details are monitored and are sent to the cloud:

- Exchange Services (Calendar, Call logs, and Contacts)
- Provisioning Server
- BroadSoft Directory
- Corporate Directory
- Ribbon Communications PAB-GAB Directory

The phone immediately sends the change in service connectivity status to the cloud. For example, if the Microsoft Exchange server gets an authentication failure, the failed authentication details are sent to cloud immediately.

If there’s no change in the service connectivity status, the phone periodically sends the status to the cloud based on the configured interval. The phone also sends the last access time of the service to the server along with response codes and failure reason if any.

Hardware Analytics

Poly phones send hardware analytics to the cloud at periodic intervals when you set the `da.supported.services` value to `all` or `hardwareanalytics`.

Poly phones send and upload the following hardware analytics and information to the cloud:

- **CPU Monitoring Service** – Sends CPU details for software processes along with total CPU consumed, Timestamp, and Monotonic time. You can set the values for trigger points such as `UpperCPUValue` and `LowerCPUValue` in percentage from the cloud. The following actions trigger the phone to send CPU details to the cloud:
  - The CPU usage value equals or goes above the `UpperCPUValue`.
  - The CPU usage value equals or goes below the `LowerCPUValue`.
  - The `UpperCPUValue` and `LowerCPUValue` are 0.

  The phone collects the records at every defined time interval. On receiving a stop command from the cloud or after timeout, the phone sends the collected records to the cloud. However, if the number of records crosses the limit of 100, the records are sent to the cloud and the counter is reset.

- **Packet Loss Service** – Uploads L2 layer network statistics (received) to the cloud through Packet Loss Service. This service has the following Rx L2 parameters:
  - `rxDiscard`
  - `rxUnicastPkts`
  - `rxBroadcastPkts`
  - `rxMulticastPkts`

  This service has the following fields:
  - `eventMonotonicTime` – Time since DUT is up.
• **uploadTime** – Time at which DUT sends the packet to the cloud.

• **versionInfo** – Every INLINE message sent to cloud contains the `versionInfo` parameter to indicate version of that message. Minor or major version change depends on type of change with respect to particular message in subsequent releases.

The following action triggers the phone to send Packet loss details to the cloud:

• Timeout
• Manually stopping service by issuing stop request

This service is applicable only for Ethernet.

• **Memory Monitoring Service** – Sends memory monitoring details for software processes along with total used, cached, and free memory to the cloud.

Memory metrics is controlled through two parameters: **UpperMemoryValue** and **LowerMemoryValue**. The following actions trigger the phone to send memory monitoring details to the cloud:

• Free memory is equal to and below **LowerMemoryValue** (Normal to Low memory)
• Free memory is equal to and above **UpperMemoryValue** (Low to Normal memory)

When you define **LowerMemoryValue** and **UpperMemoryValue** as 0, memory information is shared with the cloud periodically.

### Device Details Sent to the Cloud

When you enable device analytics, the phones can send various details regarding the device to the cloud service.

#### Device Asset Details

Device asset details include details for a primary device and SIP service. A primary device consists of Poly phones, and a secondary device consists of Bluetooth or USB headsets, expansion modules (if supported), connected cameras, and a PC port.

When you enable device analytics, the phone sends the following primary device details to the cloud:

• Manufacturer
• Product Family
• Power Source
• MAC Address
• PCS Number
• PCS Account Code
• Region Code
• Version Information
• Hardware Model
• Hardware Revision
• Hardware Part number
• Serial Number
• Obi Number
• Offset GMT
• Reboot Type
• Mac Address
• Software Release
• Upload Time
• Updater Version

Secondary Device Details
When you connect a secondary device to a Poly phone and enable device analytics with the parameter `da.supported.services` value set as `all` or `sdi`, the secondary device details are sent to the cloud. The following table lists six secondary devices and the device details they send to the cloud.

<table>
<thead>
<tr>
<th>Bluetooth Headset</th>
<th>USB Headset</th>
<th>Expansion Module</th>
<th>PC Port</th>
<th>Polycom EagleEye Mini USB Camera</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Connection Type</td>
<td>• Display Name</td>
<td>• Display Name</td>
<td>• Mac Address</td>
<td>• Connection Type</td>
</tr>
<tr>
<td>• Peripheral Type</td>
<td>• Connection Type</td>
<td>• Peripheral Type</td>
<td>• Power Source</td>
<td>• Display Name</td>
</tr>
<tr>
<td>• Display Name</td>
<td>• Peripheral Type</td>
<td>• Serial Number</td>
<td></td>
<td>• Connection Type</td>
</tr>
<tr>
<td>• Bluetooth Address</td>
<td>• Power Source</td>
<td>• Peripheral Type</td>
<td></td>
<td>• Display Name</td>
</tr>
</tbody>
</table>

Service Details
When you enable device analytics and set the `da.supported.services` parameter value to `all` or `service`, the phone sends the following SIP service details to the cloud:

• Registration Type
• SIP Server Address
• SIP User Registration Address
• SIP User ID
• Transport Protocol
• SIP Port
• Outbound Proxy Address
• Outbound Proxy Transport Protocol
• Outbound Proxy Port
• Line Type
• Display Name
• Registration Status
Device Network Details

When the phone’s network boots up or when there’s a change in network parameters, the phone sends device network details to Polycom Cloud Services.

Poly phones send network information for the Ethernet to the cloud when the phone is idle and send Wi-Fi information to the cloud at any time.

When you enable device analytics and set the `da.supported.services` parameter value to `all` or `ni`, the phone sends the following device network details for Ethernet to the cloud:

- Connection Type
- IPv4 Address
- IPv4 Subnet
- IPv4 Gateway
- VLAN
- IPv4 Address Source
- IPv6 Global Address
- Interface Name
- IPv6 Address Source
- IPv6 Link Local Address
- IPv6 ULA
- DNS Primary Address
- DNS Alternative Address
- DNS Domain
- Connection Speed
- PC Port Status
- LLDP Status
- LLDP Neighbors
- LLDP Location Information
- CDP Status
- 802.1x Status
- NTP Server
- EAP Method
- Provisioning Protocol
- Connection Mode

When Poly phones are connected to a wireless network, the phones send the following network details for the wireless network to the cloud:

- IPv4 Subnet
• Upload Time
• Version Information
• Wifi Channel
• Connection Type
• Regulatory Domain
• IPv4 Address
• IPv4 Gateway
• DNS Primary Address
• DNS Alternative Address
• Interface Name
• IPv4 Address Source
• DNS Domain
• EAP Method
• Provisioning Protocol
• MIC Error Count
• EAP Error Count
• NTP Server

Call Experience Details
When you enable device analytics on your phone and set the `da.supported.services` parameter value to `all` or `vqmon` along with the dependent features Voice Quality Monitoring Reports (vqmon) and RTP Control Protocol Extended Reports (RTCP XR), the phone sends the following details of Voice Quality Monitoring Reports to the cloud during and after the end of each call:

• Voice Quality Report Type
• Start/Stop Timestamps
• Jitter Buffer
• Packet Loss
• Session Description
• Burst Gap Loss
• Quality Estimate
• Signal Metrics
• Delay Metrics
• Remote Tag
• Local Tag
• Call-ID

Call Data Record (CDR)
When the phone ends an active call and you set the `da.supported.services` parameter value to `all` or `cdr`, the phone sends following call summary details to the cloud:

• User
• Remote Party
• Call Direction
• Disconnect Information
• Start Time
• Call Duration
• Protocol Type
• Call Rate
• Call ID
• Remote Tag
• Local Tag
• Obi number

Device Diagnostics Details

Poly phones can send device diagnostics details to the cloud, and you can perform diagnostic actions such as restart, reboot, factory reset and check synchronization from the cloud.

When you enable this option, the phone sends the following details to the cloud:

- Core dump file – Sent to the cloud when you set the `da.supported.services` parameter value to `all` or `core`.
- TSID file – Sent to the cloud when you set the `da.supported.services` parameter value to `all` or `tsid`.

Diagnostic Details for System Logs

When you enable device analytics and set the `da.supported.services` parameter value to `all` or `log`, the phone sends the system log details to the cloud.

When the phone receives a start command from the cloud, the phone sets the value for the `log.render.level`, `log file size`, and `timeout` parameters to the value in the command and starts capturing logs.

The phone uploads log files to the cloud recursively in any of the following cases:

- The file size reaches the threshold limit (configured through start command).
- The phone receives a stop command from the cloud, which resets the `log.render.level` parameter to the previously configured value on the phone.
- The system times out (configured through start command).

Set the file size as well as different log levels appropriately (as per debugging requirements) to avoid excess log capturing; otherwise, it might result in the generation of too many log files uploading to the cloud. This can also impact the phone’s efficiency. For example, set the file size threshold limit to approximately 50 KB to debug for core or user interface issues. If you want to set many module log parameters to the debug log level, set the threshold limit to 64 KB or higher.
Diagnostic Details for Packet Capture

Set the `da.supported.services` parameter value to `all` or `pcap` to capture the desired packets. Packet capture starts after receiving a request from the cloud, and the phone sends the captured files to the cloud periodically in any of the following ways:

- Until the timeout occurs
- On receiving a stop request from the cloud
- On expiry of the upload interval or max packets limit reached

**Note:** When you enable or use the remote packet capture feature, the packet capture (started through cloud) stops and the captured packets are uploaded to the cloud.

Set the filter properly to avoid excess packet capturing file creation; otherwise, too many files will be generated and uploaded to the cloud, and this might impact the phone’s efficiency. Enable the PCAP feature and set the parameter `work.diags.pcap.enabled` value to 1 to receive PCAP requests from the cloud.

When you set the packet capture filter, follow the filter convention. The following table lists the supported PCAP filter strings.

**Supported PCAP Filter Strings**

<table>
<thead>
<tr>
<th>Filter Strings/Types</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>tcp port &lt;number&gt;</code></td>
<td><code>udp port 5060 //udp sip traffic</code></td>
</tr>
<tr>
<td><code>port &lt;number&gt;</code></td>
<td><code>port 5060 //both udp/tcp traffic on port 5060</code></td>
</tr>
<tr>
<td><code>portrange &lt;startportnumber-endportnumber&gt;</code></td>
<td><code>port 80 //http traffic</code></td>
</tr>
<tr>
<td></td>
<td>`(port 5060)</td>
</tr>
<tr>
<td></td>
<td><code>portrange 200-300 //traffic on ports from 200 to 300.</code></td>
</tr>
<tr>
<td><code>dst host &lt;host&gt;</code></td>
<td><code>dst host 1.1.1.1</code></td>
</tr>
<tr>
<td><code>src host &lt;host&gt;</code></td>
<td><code>dst host www.xyz.com</code></td>
</tr>
<tr>
<td><code>host &lt;host&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>ip, arp, icmp</code></td>
<td><code>ip //all ipv4 packets</code></td>
</tr>
<tr>
<td><code>ether host &lt;MAC Address&gt;</code></td>
<td><code>ether host 1.1.1.1.1</code></td>
</tr>
<tr>
<td><code>ether proto &lt;ethernet-type&gt;</code></td>
<td><code>arp //all arp packets</code></td>
</tr>
<tr>
<td></td>
<td><code>ether proto 0x800 //For ipv4 packet filtering</code></td>
</tr>
<tr>
<td></td>
<td><code>ether proto 0x806 //For arp packet filtering</code></td>
</tr>
<tr>
<td><code>vlan &lt;id&gt;</code></td>
<td><code>vlan 6</code></td>
</tr>
</tbody>
</table>
### Filter Strings/Types

<table>
<thead>
<tr>
<th>Expression</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>dst net &lt;net&gt;</code></td>
<td><code>dst net 10.10</code> //All packets with destination subnet as - 10.10.0.0/16</td>
</tr>
<tr>
<td><code>src net &lt;net&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>net &lt;net&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>greater &lt;packet-length&gt;</code></td>
<td><code>greater 500</code> //All packets with size &gt;= 500</td>
</tr>
<tr>
<td><code>less &lt;packet-length&gt;</code></td>
<td></td>
</tr>
</tbody>
</table>
| Logical expressions (with `and`/`or`/`&`/`|`) | `ether[0] & 1 = 0` ...

### Device Analytics Parameters

Use the following parameters to configure device analytics. You can configure the device analytics feature to only enable services of your choice.

**feature.da.enabled**

- 0 (default) - Disable device analytics.
- 1 - Enable device analytics.
  - Change causes system to restart or reboot.

**feature.obitalk.enabled**

- 0 (default) - Disable the connection to the OBiTalk cloud.
- 1 - Enable the connection to the OBiTalk cloud.
- Change causes system to restart or reboot.

**obitalk.accountCode**

- Null (default)
- String (maximum of 256 characters).
- Change causes system to restart or reboot.

**da.supported.services**

- Specify the device analytics service to enable.
- all (default)
- Configure the following strings (maximum of 2048 characters) using a comma-separated list.
- ni
- service
- tsid
pcap
log
config
core
vqmon
cdr
uptimeanalytics
hardwareanalytics
uianalytics
restart
reboot
resettofactory
restapi
Change causes system to restart or reboot.

deviceAnalytics.note
Sets the self-note value on the phone and sends to cloud with primary device information message.
Null (default)
String (maximum of 512 characters).

Cloud Service Commands
The following table depicts the guiding value parameter validation for commands received from cloud:

<table>
<thead>
<tr>
<th>Service</th>
<th>Cloud Command</th>
<th>Field</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory Monitoring</td>
<td>START_INLINE</td>
<td>Timeout</td>
<td>1 min</td>
<td>1440 min</td>
<td>60 min</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Interval</td>
<td>60 sec</td>
<td>86400 sec</td>
<td>300 sec</td>
</tr>
<tr>
<td>CPU Monitoring</td>
<td>START_INLINE</td>
<td>Timeout</td>
<td>1 min</td>
<td>1440 min</td>
<td>5 min</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Interval</td>
<td>1 sec</td>
<td>60 sec</td>
<td>10 sec</td>
</tr>
<tr>
<td>Packet Loss Monitoring</td>
<td>START_INLINE</td>
<td>Timeout</td>
<td>1 min</td>
<td>1440 min</td>
<td>60 min</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Interval</td>
<td>30 sec</td>
<td>86400 sec</td>
<td>60 sec</td>
</tr>
<tr>
<td>PCAP</td>
<td>START_UPLOAD</td>
<td>BufferSize</td>
<td>3000 packets</td>
<td>5000 packets</td>
<td>5000 packets</td>
</tr>
<tr>
<td>Service</td>
<td>Cloud Command</td>
<td>Field</td>
<td>Minimum</td>
<td>Maximum</td>
<td>Default</td>
</tr>
<tr>
<td>-----------------</td>
<td>---------------</td>
<td>---------</td>
<td>---------</td>
<td>-----------</td>
<td>-----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Timeout</td>
<td>60 sec</td>
<td>86400 sec</td>
<td>600 sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Interval</td>
<td>180 sec</td>
<td>same as timeout value</td>
<td>180 sec</td>
</tr>
<tr>
<td>BLF</td>
<td>UPDATE_INLINE</td>
<td>Interval</td>
<td>5 min</td>
<td>1440 min</td>
<td>360 min</td>
</tr>
<tr>
<td>SCA</td>
<td>UPDATE_INLINE</td>
<td>Interval</td>
<td>5 min</td>
<td>1440 min</td>
<td>360 min</td>
</tr>
<tr>
<td>UI Analytics</td>
<td>UPDATE_INLINE</td>
<td>Interval</td>
<td>10 min</td>
<td>1440 min</td>
<td>360 min</td>
</tr>
<tr>
<td>Uptime Analytics</td>
<td>UPDATE_INLINE</td>
<td>Interval</td>
<td>5 min</td>
<td>1440 min</td>
<td>15 min</td>
</tr>
</tbody>
</table>
# Troubleshooting

## Topics:
- Updater Error Messages and Possible Solutions
- Polycom UC Software Error Messages
- Network Authentication Failure Error Codes
- Power and Start-up Issues
- Screen and System Access Issues
- Calling Issues
- Display Issues
- Audio Issues
- Software Upgrade Issues
- Provisioning Issues
- Factory Reset Your Phone (Hard Reboot)

The following sections address issues you might encounter when configuring phones, along with suggested actions to resolve them.

## Updater Error Messages and Possible Solutions

If a fatal error occurs, the phone doesn’t boot up.

If the error isn’t fatal, the phone boots up but its configuration might be changed. Most updater errors are logged to the phone’s boot log. However, if the phone is having trouble connecting to the provisioning server, the phone is not likely to upload the boot log.

The following table describes possible solutions to updater error messages.

<table>
<thead>
<tr>
<th>Error Message</th>
<th>Cause and Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failed to get boot parameters via DHCP</td>
<td>The phone doesn’t have an IP address and therefore can’t boot.</td>
</tr>
<tr>
<td></td>
<td>• Check that all cables are connected, the DHCP server is running, and that the phone has not been set to a VLAN that is separate from the DHCP server.</td>
</tr>
<tr>
<td></td>
<td>• Check the DHCP configuration.</td>
</tr>
<tr>
<td>Error Message</td>
<td>Cause and Possible Solution</td>
</tr>
<tr>
<td>----------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Application <file name> is not compatible with this phone!                  | An application file was downloaded from the provisioning server, but it cannot be installed on this phone.  
Install a compatible software image on the provisioning server. Be aware that there are various hardware and software dependencies. |
| Could not contact boot server using existing configuration                   | The phone cannot contact the provisioning server. Possible causes include:  
• Cabling issues  
• DHCP configuration  
• Provisioning server problems  
The phone can recover from this error so long as it previously downloaded a valid application BootROM image and all of the necessary configuration files. |
| Error, application is not present!                                         | The phone does not have an application stored in device settings and cannot boot because an application could not be downloaded.  
• Download compatible Polycom UC Software to the phone using one of the supported provisioning protocols.  
If no provisioning server is configured on the phone, enter the provisioning server details after logging in to the Updater menu and navigating to the Provisioning Server menu. |

**Polycom UC Software Error Messages**

If an error occurs in the UC Software, an error message and a warning icon displays on the phone. The following table describes Polycom UC Software error messages.
### Polycom UC Software Error Messages

<table>
<thead>
<tr>
<th>Error Message</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>Config file error: Files contain invalid params: <code>&lt;filename1&gt;</code>, <code>&lt;filename2&gt;</code>,...</td>
<td>These messages display if the configuration files contain these deprecated parameters:</td>
</tr>
<tr>
<td>Config file error: <code>&lt;filename&gt;</code> contains invalid params</td>
<td>• <code>tone.chord.ringer.x.freq.x</code></td>
</tr>
<tr>
<td>The following contain pre-3.3.0 params: <code>&lt;filename&gt;</code></td>
<td>• <code>se.pat.callProg.x.name</code></td>
</tr>
<tr>
<td></td>
<td>• <code>ind.anim.IP_500.x.frame.x.duration</code></td>
</tr>
<tr>
<td></td>
<td>• <code>ind.pattern.x.step.x.state</code></td>
</tr>
<tr>
<td></td>
<td>• <code>feature.2.name</code></td>
</tr>
<tr>
<td></td>
<td>• <code>feature.9.name</code></td>
</tr>
<tr>
<td></td>
<td>This message also displays if any configuration file contains more than 100 of the following errors:</td>
</tr>
<tr>
<td></td>
<td>• Unknown parameters</td>
</tr>
<tr>
<td></td>
<td>• Out-of-range values</td>
</tr>
<tr>
<td></td>
<td>• Invalid values.</td>
</tr>
<tr>
<td></td>
<td>To check that your configuration files use correct parameter values, refer to Using Correct Parameter XML Schema, Value Ranges, and Special Characters.</td>
</tr>
<tr>
<td>Line: Unregistered</td>
<td>This message displays if a line fails to register with the call server.</td>
</tr>
<tr>
<td>Login credentials have failed. Please update them if information is incorrect.</td>
<td>This message displays when the user enters incorrect login credentials on the phone: Status &gt; Basic &gt; Login Credentials.</td>
</tr>
<tr>
<td>Missing files, config. reverted</td>
<td>This message displays when errors in the configuration and a failure to download the configuration files force the phone to revert to its previous (known) condition with a complete set of configuration files. This also displays if the files listed in the <code>&lt;MAC Address&gt;.cfg</code> file are not present on the provisioning server.</td>
</tr>
<tr>
<td>Network link is down</td>
<td>Indicates that the phone cannot establish a link to the network and persists until the link problem is resolved. Call-related functions, and phone keys are disabled when the network is down but the phone menu works.</td>
</tr>
</tbody>
</table>

### Network Authentication Failure Error Codes

Error messages display on the phone if 802.1X authentication fails.

The error codes display on the phone when you press the **Details** key. Error codes are also included in the log files.
<table>
<thead>
<tr>
<th>Event Code</th>
<th>Description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Unknown events</td>
<td>An unknown event by '1' can include any issues listed in this table.</td>
</tr>
<tr>
<td>2</td>
<td>Mismatch in EAP Method type</td>
<td>Authenticating server's list of EAP methods does not match with clients'.</td>
</tr>
</tbody>
</table>
| 30xxx      | TLS Certificate failure      | 000 - Represents a generic certificate error. The phone displays the following codes:  
|            |                              | • 042 - bad cert  
|            |                              | • 043 - unsupported cert  
|            |                              | • 044 - cert revoked  
|            |                              | • 045 - cert expired  
|            |                              | • 046 - unknown cert  
|            |                              | • 047 - illegal parameter  
|            |                              | • 048 - unknown CA  
| 31xxx      | Server Certificate failure   | 'xxx' can use the following values:  
|            |                              | • 009 - Certificate not yet Valid  
|            |                              | • 010 - Certificate Expired  
|            |                              | • 011 - Certificate Revocation List  
|            |                              | • (CRL) not yet Valid  
|            |                              | • 012 - CRL Expired  
| 4xxx       | Other TLS failures           | 'xxx' is the TLS alert message code. For example, if the protocol version presented by the server is not supported by the phone, then 'xxx' is 70, and the EAP error code is 4070. See section 7.2 of RFC 2246 for further TLS alert codes and error codes. |
| 5xxx       | Credential failures          | 5xxx - wrong user name or password                                        |
| 6xxx       | PAC failures:                |  
|            |                              | • 080 - No PAC file found  
|            |                              | • 081 - PAC file password not provisioned  
|            |                              | • 082 - PAC file wrong password  
|            |                              | • 083 - PAC file invalid attributes  

Troubleshooting
### Power and Start-up Issues

The following table describes possible solutions to power and start-up issues.

<table>
<thead>
<tr>
<th>Power or Start-up Issue</th>
<th>Possible Solutions:</th>
</tr>
</thead>
<tbody>
<tr>
<td>The phone has power issues or the phone has no power.</td>
<td>Determine whether the problem is caused by the phone, the AC outlet, or the PoE switch. Do one of the following:</td>
</tr>
<tr>
<td></td>
<td>• Verify that no lights appear on the unit when it is powered up.</td>
</tr>
<tr>
<td></td>
<td>• Check to see if the phone is properly plugged into a functional AC outlet.</td>
</tr>
<tr>
<td></td>
<td>• Make sure that the phone is not plugged into an outlet controlled by a light switch that is turned off.</td>
</tr>
<tr>
<td></td>
<td>• If the phone is plugged into a power strip, try plugging directly into a wall outlet instead.</td>
</tr>
<tr>
<td>The phone does not boot.</td>
<td>If the phone does not boot, there may be a corrupt or invalid firmware image or configuration on the phone:</td>
</tr>
<tr>
<td></td>
<td>• Ensure that the provisioning server is accessible on the network and a valid software load and valid configuration files are available.</td>
</tr>
<tr>
<td></td>
<td>• Ensure that the phone is configured with the correct address for the provisioning server on the network.</td>
</tr>
</tbody>
</table>
## Screen and System Access Issues

The following table describes possible solutions to screen and system access issues.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Cause and Possible Solution</th>
</tr>
</thead>
</table>
| There is no response from feature key presses. | If your phone keys do not respond to presses:  
  • Press the keys more slowly.  
  • Check to see whether or not the key has been mapped to a different function or disabled.  
  • Make a call to the phone to check for inbound call display and ringing. If successful, try to press feature keys while a call is active to access a directory or buddy status.  
  • On the phone, go to **Menu > Status > Lines** to confirm the line is actively registered to the call server.  
  Reboot the phone to attempt re-registration to the call server. Go to **Menu > Settings > Advanced > Reboot Phone**. |
| The display shows the message "Network Link is Down". | This message displays when the LAN cable is not properly connected. Do one of the following:  
  • Check the termination at the switch or hub end of the network LAN cable.  
  • Check that the switch or hub is operational (flashing link/status lights).  
  • On the phone, go to **Menu > Status > Network**. Scroll down to verify that the LAN is active.  
  • Ping the phone from a computer.  
  Reboot the phone to attempt re-registration to the call server. Go to **Menu > Settings > Advanced > Reboot Phone**. |

## Calling Issues

The following table provides possible solutions to common calling issues.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Cause and Possible Solution</th>
</tr>
</thead>
</table>
| There is no dial tone. | If there is no dial tone, power may not be correctly supplied to the phone. Try one of the following:  
  • Check that the display is illuminated.  
  • Make sure the LAN cable is inserted properly at the rear of the phone; try unplugging and re-inserting the cable.  
  If you are using in-line powering, check that the switch is supplying power to the phone. |
The phone does not ring.
If there is no ringtone but the phone displays a visual indication when it receives an incoming call, do the following:
• Adjust the ring level from the front panel using the volume up/down keys.
Check the status of handset, headset (if connected), and hands-free speakerphone.

The line icon shows an unregistered line icon.
If the phone displays an icon indicating that a line is unregistered, re-register the line and place a call.

Display Issues
The following table provides tips for resolving display screen issues.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Cause and Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>There is no display or the display is incorrect.</td>
<td>If there is no display, power may not be correctly supplied to the phone. Do one of the following:</td>
</tr>
</tbody>
</table>
|                                       | • Check that the display is illuminated.
|                                       | • Make sure the power cable is inserted properly at the rear of the phone.
|                                       | • If you are using PoE powering, check that the PoE switch is supplying power to the phone.                                                                                                                                 |
|                                       | Use the screen capture feature to verify whether the screen displays properly in the capture. Refer to Capture Your Device's Current Screen.                                                                                       |
| The display is too dark or too light. | The phone contrast may be set incorrectly. Do one of the following:                                                                                                                                                           |
|                                       | • Adjust the contrast.
|                                       | • Reboot the phone to obtain the default level of contrast.                                                                                                                                                                    |
| The display is flickering.            | Certain types of older fluorescent lighting may cause the display to flicker. If your phone is in an environment with fluorescent lighting, angle or move the Polycom phone away from the lights.                                      |
| The time and date are flashing.      | If the time and date are flashing, the phone is disconnected from the LAN or there is no SNTP time server configured. Do one of the following:                                                                                       |
|                                       | • Reconnect the phone to the LAN.
|                                       | • Configure an SNTP server.                                                                                                                                                                                                    |
|                                       | Disable the time and date if you do not want to connect your phone to a LAN or SNTP server.                                                                                                                                     |
Audio Issues
The following table describes possible solutions to audio issues.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Cause and Possible Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>There is no audio on the headset</td>
<td>If there is no audio on your headset, the connections may not be correct. Do one of the following:</td>
</tr>
<tr>
<td></td>
<td>• Ensure the headset is plugged into the jack marked Headset at the rear of the phone.</td>
</tr>
<tr>
<td></td>
<td>• Ensure the headset amplifier (if present) is turned on and adjust the volume.</td>
</tr>
</tbody>
</table>

Software Upgrade Issues
The following table describes possible solutions to issues that may occur during or after a software upgrade.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Cause and Possible Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Some settings or features are not working as expected on the phone.</td>
<td>The phone's configuration may be incorrect or incompatible. Check for errors on the phone by navigating to Menu &gt; Status &gt; Platform &gt; Configuration. If there are messages stating Errors Found, Unknown Params, or Invalid values, correct your configuration files and restart the phone.</td>
</tr>
<tr>
<td>The phone displays a Config file error message for five seconds after it boots up.</td>
<td>You are using configuration files from a UC Software version earlier than the UC Software image running on the phones. Configuration parameters and values can change each release and specific parameters may or may not be included. See the UC Software Administrator's Guide and Release Notes for the UC Software version you have installed on the phones. Correct the configuration files, remove the invalid parameters, and restart the phone.</td>
</tr>
</tbody>
</table>
Issue | Cause and Possible Solutions
--- | ---
When using the Web Configuration Utility to upgrade phone software, the phone is unable to connect to the Polycom Hosted Server. | Occasionally, the phone is unable to connect to the Polycom-hosted server because of the following:
• The Polycom-hosted server is temporarily unavailable.
• There is no software upgrade information for the phone to receive.
• The network configuration is preventing the phone from connecting to the Polycom hosted server.

To troubleshoot the issue:
• Try upgrading your phone later.
• Verify that new software is available for your phone using the Polycom UC Software Release Matrix.
• Verify that your network’s configuration allows the phone to connect to [http://downloads.polycom.com](http://downloads.polycom.com).

If the issue persists, try manually upgrading your phone's software.

### Provisioning Issues

If settings you make from the central server aren’t working, check first for priority settings applied from the phone menu system or system web interface. Afterward, check for duplicate settings in your configuration files.

### Factory Reset Your Phone (Hard Reboot)

**Symptom:**

The phone gets into a state where you must factory reset the phone, but you can’t access the phone’s local interface to reset the phone to default settings. You also can’t access the factory reset option through the system web interface.

**Workaround**

You can factory reset your phone using the phone’s hardware (hard reboot). Resetting the phone to defaults clears the flash parameters, removes user and cached data, and resets the administrator password.

**Note:** It may take several tries to get the timing right or to find the correct spots to press on the LCD display.

**Procedure**

1. Disconnect the power, then power on the Poly phone.
2. As soon as the Poly logo shows on the screen, press and hold the four corners of the LCD display.
3. Release the LCD display when the **Mute** indicator on the lower-right corner of the phone begins flashing red, amber, and green.
System Logs

Topics:
- Configuring Log Files
- Logging Levels
- Uploading Logs to a USB Flash Drive

System log files assist when troubleshooting issues.

System log files contain information about system activities and the system configuration profile. After you set up system logging, you can retrieve system log files.

The detailed technical data in the system log files can help Poly Global Services resolve problems and provide technical support for your system. Your support representative may ask you to download log archives and send them to Poly Global Services.

You must contact Poly Customer Support to obtain the template file (`techsupport.cfg`) that contains the parameters that configure log levels.

Configuring Log Files

You can configure log files using logging parameters.

Log file names use the following format: `[MAC address]_[Type of log].log`. For example, if the MAC address of your phone is `0004f2203b0`, the app log file name is `0004f2203b0_app.log`.

The phone writes information into several different log files. The following list describes the type of information in each type of log file.

- **Boot Log** – Boot logs are sent to the provisioning server in a boot.log file collected from the Updater/BootROM application each time the phone boots up. The BootROM/Updater application boots the application and updates with the new firmware if available.
- **Application Log** – The application log file contains complete phone functionality including SIP signaling, call controls and features, digital signal processor (DSP), and network components.
- **Syslog** – For more information about Syslog, see [Syslog on Polycom Phones - Technical Bulletin 17124](#).

Severity of Logging Event Parameter

You can configure the severity of the events that are logged independently for each module of the UC Software.

This enables you to capture lower severity events in one part of the application, and high severity events for other components. Severity levels range from 0 to 6, where 0 is the most detailed logging and 6 captures only critical errors.

**Note:** User passwords display in level 1 log files.

You must contact Poly Customer Support to obtain the template file `techsupport.cfg` containing parameters that configure log levels.
log.level.change.module_name

Set the severity level to log for the module name you specify. Not all modules are available for all phone models.

For a list of available module names, module descriptions, and log level severity, see refer to the Web Configuration Utility at Settings > Logging > Module Log Level Limits.

Log File Collection and Storage Parameters

You can configure log file collection and storage using the parameters in the following list.

You must contact Customer Support to obtain the template file techsupport.cfg containing parameters that configure log file collection and storage.

There is no way to prevent the system log file [MAC address]-plcmsyslog.tar.gz from uploading to the server and you cannot control it using the parameters log.render.file.upload.append.sizeLimit and log.render.file.upload.append.limitMode. However, you can control the frequency of uploads using log.render.file.upload.system.period.

log.render.level

Specify the events to render to the log files. Severity levels are indicated in brackets.

0 - SeverityDebug (7)
1 - SeverityDebug (7) - default
2 - SeverityInformational (6)
3 - SeverityInformational (6)
4 - SeverityError (3)
5 - SeverityCritical (2)
6 - SeverityEmergency (0)

log.render.file.size

Set the maximum file size of the log file. When the maximum size is about to be exceeded, the phone uploads all logs that have not yet been uploaded and erases half of the logs on the phone. You can use a web browser to read logs on the phone.

512 kb (default)

log.render.file.upload.period

Specify the frequency in seconds between log file uploads to the provisioning server.

Note: The log file is not uploaded if no new events have been logged since the last upload.

172800 seconds (default) - 48 hours

log.render.file.upload.append

1 (default) - Log files uploaded from the phone to the server are appended to existing files. You must set up the server to append using HTTP or TFTP.

0 - Log files uploaded from the phone to the server overwrite existing files.
Note that this parameter is not supported by all servers.

**log.render.file.upload.append.sizeLimit**

Specify the maximum size of log files that can be stored on the provisioning server.

- 512kb (default)

Note that this parameter is not supported by HTTP/HTTPS or TFTP protocols. Logs generated and uploaded via HTTP/HTTPS or TFTP protocol must be deleted manually if needed.

**log.render.file.upload.append.limitMode**

Specify whether to stop or delete logging when the server log reaches its maximum size.

- delete (default) - Delete logs and start logging again after the file reaches the maximum allowable size specified by `log.render.file.upload.append.sizeLimit`.
- stop - Stop logging and keep the older logs after the log file reaches the maximum allowable size.

Note that this parameter is not supported by HTTP/HTTPS or TFTP protocols. Logs generated and uploaded via HTTP/HTTPS or TFTP protocol must be deleted manually if needed.

**Scheduled Logging Parameter**

Scheduled logging can help you monitor and troubleshoot phone issues.

Use the parameters in this list to configure scheduled logging.

You must contact Polycom Customer Support to obtain the template file `techsupport.cfg` containing parameters that configure scheduled logging.

**log.sched.x.name**

Configure the number of debug commands you want to schedule an output for. You can configure 1-10 debug commands per phone. Set the number of debug commands as x.

- If x = 1, the default command name is 'showCpuLoad'.
  - 9 (default)
- If x = 2, the default command name is 'showBatteryStat'.
  - 22 (default)
- 3 - 10 = No default value

The following are permitted values:

- NULL
- memShow
- checkStack
- cameraLogShow
- ls
- ifShow
- ifShowVerbose
- showProcesses
If you encounter any camera related issue, set the log.sched.x.name value to cameraLogShow where x = 1 or 2 and set log.level.change.slog=2.

Logging Levels

The event logging system supports the classes of events listed in the table Logging Levels.

Two types of logging are supported:

- Level, change, and render
- Schedule

Note: Logging parameter changes can impair system operation. Do not change any logging parameters without prior consultation with Technical Support.

<table>
<thead>
<tr>
<th>Logging Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Debug only</td>
</tr>
<tr>
<td>1</td>
<td>High detail class event</td>
</tr>
<tr>
<td>2</td>
<td>Moderate detail event class</td>
</tr>
<tr>
<td>3</td>
<td>Low detail event class</td>
</tr>
<tr>
<td>4</td>
<td>Minor error</td>
</tr>
<tr>
<td>Logging Level</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>5</td>
<td>Major error – will eventually incapacitate the system</td>
</tr>
<tr>
<td>6</td>
<td>Fatal error</td>
</tr>
</tbody>
</table>

Each event in the log contains the following fields separated by the pipe (|) character:
- Time or time/date stamp, in one of the following formats:
  - 0 - milliseconds \(011511.006 = 1\) hour, 15 minutes, 11.006 seconds since booting
  - 1 - absolute time with minute resolution \(0210281716 - 2002\) October 28, 17:16
  - 2 - absolute time with seconds resolution \(1028171642 - 2002\) October 28, 17:16:42
- 1-5 character component identifier (such as "so")
- Event class
- Cumulative log events missed due to excessive CPU load
- The event description

**Logging Level, Change, and Render Parameters**

The following list includes parameters for configuring logging features.

**log.level.change.xxx**
Controls the logging detail level for individual components. These are the input filters into the internal memory-based log system.

4 (default)

0 - 6

Possible values for xxx are acom, ares, app1, bluet, bdiag, brow, bsdir, cap, cdp, cert, cfg, cipher, clink, clist, cmp, cmr, copy, curl, daa, dapi, dasvc, dbs, dbuf, dhcpc, dis, dock, dot1x, dns, drvbt, ec, efk, ethf, flk, fec, fecde, fecen, fur, hset, httpa, httpd, hw, ht, ib, key, ldap, lic, lldp, loc, log, mb, mcu, mobil, mrci, net, niche, ocsmp, osd, pcap, pcd, pdc, peer, pgui, pkt, pmt, poll, pps, pres, pstn, ptt, push, pwrsv, rdisk, res, restapi, rtos, rtl, sec, sig, sip, slog, so, srtp, sshc, ssps, statc, statn, style, sync, sys, ta, task, tls, trace, ttrs, usb, usbio, util, utilm, vsr, wdog, wmgr, and xmpp.

**log.level.change.fec**
Sets the log level for video FEC.

4 (default)

0 - 6

**log.level.change.fecde**
Sets high volume log level to decode video FEC.

4 (default)

0 - 6
**log.level.change.fecen**
Sets high volume log level to encode video FEC.
4 (default)
0 - 6

**log.level.change.flk**
Sets the log level for the FLK logs.
4 (default)
0 - 6

**log.level.change.prox**
Initial logging level for the Proximity log module.
4 (default)
0 - 6

**log.level.change.ptp**
Initial logging level for the Precision Time Protocol log module.
4 (default)
0 - 6

**log.level.change.sopi**
Specify the SOPI service log level for the Ribbon Communications Global Address Book and Personnel Address Book.
4 (default)
0 - 6

**log.render.file**
When you enable this option, the phone first writes log files directly into its flash memory. The contents of the flash memory then upload to a provisioning server after a predetermined period of time or when the flash memory becomes full.
1 (default) - The phone uploads the log file content to the server.
0 - The phone prevents uploading the log file content to the server.

**Note:** Poly recommends that you prevent the ability to upload log files only when necessary to reduce data traffic when the phone starts or reboots.

**log.render.realtime**
Poly recommends that you do not change this value.
1 (default) - Enable
0 - Disable
log.render.stdout

Poly recommends that you do not change this value.
0 (default) - Disable
1 - Enable

log.render.type

Refer to the Event Timestamp Formats table for timestamp type.
2 (default)
0 - 2

Logging Parameters

The phone can be configured so certain advanced logging tasks take place scheduled basis.
Poly recommends that you set the parameters listed below with consultation with Polycom Technical Support. Each scheduled log task is controlled by a unique parameter set starting with log.sched.x where x identifies the task. A maximum of 10 schedule logs is allowed.

log.sched.x.level

The event class to assign to the log events generated by this command.
3 (default)
0 - 5
This needs to be the same or higher than log.level.change.slog for these events to display in the log.

log.sched.x.period

Specifies the time in seconds between each command execution.
15 (default)
positive integer

log.sched.x.startDay

When startMode is abs, specifies the day of the week to start command execution. 1=Sun, 2=Mon, ..., 7=Sat
7 (default)
0 - 7

log.sched.x.startMode

Starts at an absolute or relative time to boot.
Null (default)
0 - 64
log.sched.x.startTime

Displays the start time in seconds since boot when startMode is rel or displays the start time in 24-hour clock format when startMode is abs.

Null (default)
positive integer, hh:mm

Uploading Logs to a USB Flash Drive

You can configure your phones to copy application and boot logs to a USB flash drive connected to the phone.

You can configure the phone to copy the application logs to the USB flash drive when the log file size reaches the limit defined in the log.render.file.size parameter. Similarly, you can configure the phone to copy application logs to the USB flash drive periodically using log.render.file.upload.period parameter.

USB Logging Parameter

The following parameters configure the USB logging feature.

feature.usbLogging.enabled

0 (default) - Disables collecting logs using a USB flash drive.
1 - Enables collecting logs using a USB flash drive.
Audio Features

Topics:

- Automatic Gain Control
- Background Noise Suppression
- Comfort Noise
- Voice Activity Detection
- Comfort Noise Payload Packets
- Synthesized Call Progress Tones
- Jitter Buffer and Packet Error Concealment
- DTMF Tones
- Acoustic Echo Cancellation
- Context-Sensitive Volume Control
- Polycom Acoustic Fence
- Location of Audio Alerts
- Ringtones
- Distinctive Ringtones
- Sound Effects
- Supported Audio Codecs
- IEEE 802.1p/Q
- IEEE 802.1p/Q Parameters
- Voice Quality Monitoring (VQMon)

After you set up your phones on the network, users can send and receive calls using the default configuration. You can configure modifications that optimize the audio quality of your network.

Poly phones support audio sound quality features and options you can configure to optimize the conditions of your organization's phone network system.

Automatic Gain Control

Automatic Gain Control (AGC) boosts the gain of the near-end conference participants and helps conference participants hear your voice.

Note: This feature is enabled by default and you can’t disable it.
Background Noise Suppression

Background noise suppression reduces the background noise caused by items such as fans, projectors, and air conditioners.

**Note:** This feature is enabled by default and you can’t disable it.

Comfort Noise

Comfort Noise ensures a consistent background noise level to provide a natural call experience.

**Note:** Comfort Noise fill isn’t related to Comfort Noise packets the phone generates when you enable Voice Activity Detection.

Voice Activity Detection

Voice activity detection (VAD) conserves network bandwidth by detecting periods of silence in the transmit data path so the phone doesn't have to transmit unnecessary data packets for outgoing audio.

For compression algorithms without an inherent VAD function, such as G.711, the phone uses the codec-independent comfort noise transmission processing specified in RFC 3389. The RFC 3389 algorithm is derived from G.711 Appendix II, which defines a comfort noise (CN) payload format (or bit stream) for G.711 use in packet-based, multimedia communication systems.

Voice Activity Detection Parameters

The following list includes the parameters you can use to configure Voice Activity Detection.

**voice.vad.signalAnnexB**

1 (default) - Annex B is used and a new line is added to SDP depending on the setting of voice.vadEnable. If voice.vadEnable is set to 1, add parameter line a=fmtp:18 annexb="yes" below a=rtpmap parameter line (where "18" could be replaced by another payload).

0 There is no change to SDP. If voice.vadEnable is set to 0, add parameter line a=fmtp:18 annexb="no" below the a=rtpmap... parameter line (where "18" could be replaced by another payload).

**voice.vadEnable**

0 - Disable Voice activity detection (VAD).
1 - Enable VAD.

**voice.vadThresh**

The threshold for determining what is active voice and what is background noise in dB. Sounds louder than this set value are considered active voice, and sounds quieter than this threshold...
are considered background noise. This does not apply to G.729AB codec operation which has its own built-in VAD function.

25 (default)
Integer from 0 - 30

Comfort Noise Payload Packets
Comfort Noise is enabled by default on Poly phones, and the payload type is negotiated in Session Description Protocol (SDP) with a default of 13 for 8 KHz codecs or 122 for 16 KHz codecs or higher.

Comfort Noise Payload Packets Parameters
The following list includes the parameters you can use to configure Comfort Noise payload packets.

voice.CNControl
Publishes support for Comfort Noise in the SDP body of the INVITE message and includes the supported comfort noise payloads in the media line for audio.

1 (default) – Either the payload type 13 for 8 KHz sample rate audio codec is sent for Comfort Noise, or the dynamic payload type for 16 KHz audio codecs are sent in the SDP body.

0 – Does not publish support or payloads for Comfort Noise.

voice.CN16KPayload
Alters the dynamic payload type used for Comfort Noise RTP packets for 16 KHz codecs.

96 to 127
122 (default)

Synthesized Call Progress Tones
Poly phones play call signals and alerts, called call progress tones, that include busy signals, ringback sounds, and call waiting tones.

The built-in call progress tones match standard North American tones. If you want to customize your phone’s call progress tones to match the standard tones in your region, contact Technical Support.

Jitter Buffer and Packet Error Concealment
Poly phones employ a high-performance jitter buffer and packet error concealment system designed to mitigate packet inter-arrival jitter and out-of-order or lost or delayed (by the network) packets.

The jitter buffer is adaptive and configurable for different network environments. When packets are lost, a concealment algorithm minimizes the resulting negative audio consequences. This feature is enabled by default.
DTMF Tones

Poly phones generate dual-tone multi-frequency (DTMF) tones, also called touch tones, in response to user dialing on the dialpad. Your phone transmits these tones in the real-time transport protocol (RTP) streams of connected calls.

Your phone can encode the DTMF tones using the active voice codec or using RFC 2833-compatible encoding. The coding format decision is based on the capabilities of the remote endpoint. The phone generates RFC 2833 (DTMF only) events but does not regenerate—or otherwise use—DTMF events received from the remote end of the call.

DTMF Tone Parameters

The following list includes the parameters you can use to set up DTMF tones.

**reg.1.telephony**

Allow telephony services for inbound and outbound calls.

1 (default) – Allowed

0 – Disallowed

**tone.dtmf.chassis.masking**

0 (default) - DTMF tones play through the speakerphone in handsfree mode.

1 - Set to 1 only if `tone.dtmf.viaRtp` is set to 0. DTMF tones are substituted with non-DTMF pacifier tones when dialing in handsfree mode to prevent tones from broadcasting to surrounding telephony devices or inadvertently transmitted in-band due to local acoustic echo.

Change causes system to restart or reboot.

**tone.dtmf.level**

The level of the high frequency component of the DTMF digit measured in dBm0; the low frequency tone is two dB lower.

-15

-33 to 3

Change causes system to restart or reboot.

**tone.dtmf.offTime**

When a sequence of DTMF tones is played out automatically, specify the length of time in milliseconds (ms) the phone pauses between digits. This is also the minimum inter-digit time when dialing manually.

50 (default)

1 – Indefinite

Change causes system to restart or reboot.
**tone.dtmf.onTime**

Set the time in milliseconds (ms) DTMF tones play on the network when DTMF tones play automatically. The time you set is also the minimum time the tone plays when manually dialing.

50 (default)

1 - 65535

Change causes system to restart or reboot.

**tone.dtmf.rfc2833Control**

Specify if the phone uses RFC 2833 to encode DTMF tones.

1 (default) - The phone indicates a preference for encoding DTMF through RFC2833 format in its Session Description Protocol (SDP) offers by showing support for the phone-event payload type. This doesn't affect SDP answers and always honor the DTMF format present in the offer.

0 - The phone doesn't offer dynamic payload for RFC2833 phone-event.

Change causes system to restart or reboot.

**tone.dtmf.rfc2833Payload**

Specify the phone-event payload encoding in the dynamic range to be used in SDP offers.

Generic (default) -127

96 to 127

Change causes system to restart or reboot.

**tone.dtmf.rfc2833Payload_OPUS**

Sets the DTMF payload required to use Opus codec.

126 (default)

96 - 127

Change causes system to restart or reboot.

**tone.dtmf.viaRtp**

1 (default) - Encode DTMF in the active RTP stream. Otherwise, DTMF may be encoded within the signaling protocol only when the protocol offers the option.

0 – If you set this parameter to 0, you must set **tone.dtmf.chassis.masking** to 1.

Change causes system to restart or reboot.

**tone.localDtmf.onTime**

Set the time in milliseconds (ms) DTMF tones play for when the phone plays out a DTML tone sequence automatically.

50 (default)

1 - 65535
tone.dtmf.rfc2833.SupportOpusClockRate

1 – (default) Publishes the Telephone-event DTMF frequency as 48000 Hz along with 8000 Hz on Opus codec.
0 - Publishes the Telephone-event DTMF frequency as 8000 Hz on Opus codec.

Acoustic Echo Cancellation

Acoustic Echo Cancellation (AEC) enables the phones to significantly reduce echo while permitting natural communication. Configure your phones to use advanced AEC for hands-free operation using the speakerphone.

The AEC feature includes the following:
- Talk State Detector: Determines whether the near-end user, far-end user, or both are speaking.
- Linear Adaptive Filter: Adaptively estimates the loudspeaker-to-microphone echo signal and subtracts that estimate from the microphone signal.
- Non-linear Processing: Suppresses any echo remaining after the Linear Adaptive Filter.

The phones also support headset echo cancellation.

Acoustic Echo Cancellation Parameters

The following list includes the parameters you can use to set up Acoustic Echo Cancellation (AEC).

voice.aec.hf.enable

1 (default) - Enables the AEC function for hands-free options.
0 - Disables the AEC function for hands-free options.

Poly doesn’t recommend disabling this parameter.

voice.aec.hs.enable

0 - Disables the AEC function for the handset.
1 (default) - Enables the AEC function for the handset.

Context-Sensitive Volume Control

In some countries, regulations state that a phone's receiver volume must be reset to a nominal level for each new call.

Transmit levels are fixed according to the TIA/EIA-810-A standard. The next list includes parameters that configure the receiver volume to reset and persist across calls each time a user makes changes to the default volume level. You can adjust the volume of phone sound effects—such as the ringer and the volume of receiving call audio—separately for the speakerphone, handset, and headset.
Context Sensitive Volume Control Parameters

The following list includes the parameters you can use to configure Context Sensitive Volume Control.

**voice.volume.persist.bluetooth.headset**
- 0 (default) - The Bluetooth headset volume does not persist between calls and resets to a nominal level each new call.
- 1 - The volume for each call is the same as the previous call.

**voice.volume.persist.handset**
- 0 (default) - The handset volume automatically resets to a nominal level after each call.
- 1 - The volume for each call is the same as the previous call.

**voice.volume.persist.handsfree**
- 1 (default) - The speakerphone volume at the end of a call persists between calls.
- 0 - The speakerphone volume does not persist between calls and resets to a nominal level each new call.

**voice.volume.persist.usb.handsfree**
- 0 (default) - Does not use USB headset automatically for calls.
- 1 - Uses the USB headset automatically for all calls.

**voice.volume.persist.usbHeadset**
- 0 (default) - The USB headset volume does not persist between calls and resets to a nominal level each new call.
- 1 - The USB headset volume at the end of a call persists between calls.

Polycom Acoustic Fence

Polycom Acoustic Fence suppresses background noise sent to the far end when you’re using:

- Phone handset
- Wired headset connected to the headset port
- USB headset connected to the phone

This feature is particularly useful in call center environments where background noise can impact far-end audio quality.

Polycom Acoustic Fence is available for the following phones and headsets:

Set `video.disableAFOnFullScreen` parameter value to 1 to optimize phone performance while using Polycom EagleEye Mini USB camera with Acoustic Fence.

**Note:** Polycom Acoustic Fence doesn't support Bluetooth headsets.
Acoustic Fence Parameters
The following list includes the parameters you can use to configure Polycom Acoustic Fence noise suppression feature.

Note: When Acoustic Fence is enabled, Polycom recommends setting the parameter `video.disableAFOnFullScreen` to 1 to improve the phone’s performance when the Polycom EagleEye Mini USB camera is connected to a CCX 600 phone.

**feature.acousticFenceUI.enabled**
- 0 (default) - Hide display of the Acoustic Fence Configuration setting on the phone.
- 1 - Displays the Acoustic Fence Configuration setting on the phone.

**voice.ns.hd.enable**
- Enables or disables noise suppression for headsets.
- 0 (default) – Disabled
- 1 – Enabled

**voice.ns.hd.enhanced**
- Enables or disables Acoustic Fence noise suppression for headsets.
- 0 (default) – Disabled
- 1 – Enabled

**voice.ns.hd.nonStationaryThresh**
- Sets the Acoustic Fence noise suppression threshold for headsets. A lower value allows more background sound to enter, and a higher value suppresses background noise. High values can suppress the speaker’s voice and impact far-end audio quality.
- 1 to 10
- 8 (default)

**voice.ns.hs.enable**
- Enables or disables noise suppression for handsets.
- 0 (default) - Disabled
- 1 - Enabled

**voice.ns.hs.enhanced**
- Enables or disables Acoustic Fence noise suppression for handsets.
- 1 (default) - Enabled
- 0 - Disabled
**voice.ns.hs.nonStationaryThresh**

Sets the Acoustic Fence noise suppression threshold for handsets. A lower value allows more background sound to enter, and a higher value suppresses background noise. High values can suppress the speaker's voice and impact far-end audio quality.

1 to 10
8 (default)

**video.disableAFOnFullScreen**

Allows or disallows the phone to dynamically deactivate Acoustic Fence when the user changes the view to full screen mode while using a handset in a video call.

0 (default) - Disallowed
1 - Allowed

**Location of Audio Alerts**

You can choose where all audio alerts, including incoming call alerts, are played on the phones.

You can specify the audio to play from the hands-free speakerphone (default), the handset, the headset, or the active location. If you choose the active location, audio alerts play out through the handset or headset if they are in use. Otherwise, alerts play through the speakerphone.

**Audio Alert Parameters**

Use the parameters in the following list to configure audio alerts and sound effects.

**se.appLocalEnabled**

Enables or disables audio alerts and sound effects.

1 (default) - Enabled
0 - Disabled

Change causes system to restart or reboot.

**se.destination**

chassis (default) - Alerts and sound effects play through the phone's speakerphone.
headset - If connected, alerts and sounds play through the headset.
handset active - Alerts play from the destination that is currently in use. For example, if a user is in a call on the handset, a new incoming call rings through the handset.

**se.stutterOnVoiceMail**

1 (default) - A stuttered dial tone is used instead of a normal dial tone to indicate that one or more voicemail messages are waiting at the message center.
0 - A normal tone is used to indicate that one or more voicemail messages are waiting at the message center.
Ringtones

Use ringtones to define a simple ring class that the phone applies based on credentials carried within the network protocol.

The ring class includes parameters such as call-waiting and ringer index (if appropriate), and it can use one of the following ring types:

- **Ring** - Plays a specified ring pattern or call waiting indication.
- **Visual** - Provides a visual indication (no audio) of an incoming call. You don’t need to specify a ringer.
- **Answer** - Provides auto-answer on an incoming call.
- **Ring-answer** - Provides auto-answer on an incoming call after a certain number of rings.

**Note:** Auto-answer for an incoming call works only when there are no other calls in progress on the phone, including other calls in progress on shared or monitored lines. However, if a phone initiates a call on a shared or monitored line, auto-answer works.

**Supported Ring Classes**

The phones support the following ring classes:

- default
- visual
- answerMute
- autoAnswer
- ringAnswerMute
- ringAutoAnswer
- internal
- external
- emergency
- precedence
- splash
- custom<y> where y is 1 to 17.

**Ringtone Parameters**

Use the following parameters to configure ringtones.

`se.rt.enabled`

Enables or disables ringtone feature.

0 - Disabled
1 (default) - Enabled
se.rt.modification.enabled
   Controls whether or not users are allowed to modify the predefined ringtone from the phone's user interface.
   0 - Users not allowed.
   1 (default) - Users allowed.

se.rt.<ringClass>.callWait
   The call waiting tone used for the specified ring class. The call waiting pattern should match the pattern defined in Call Progress Tones.
   callWaiting (default)
   callWaitingLong
   precedenceCallWaiting

se.rt.<ringClass>.name
   The answer mode for a ringtone, which is used to identify the ringtone in the user interface.
   UTF-8 encoded string

se.rt.<ringClass>.ringer
   The ringtone used for this ring class. The ringer must match one listed in Ringtones.
   default
   ringer1 to ringer24
   ringer2 (default)

se.rt.<ringClass>.timeout
   The duration of the ring in milliseconds before the call is auto-answered, which only applies if the type is set to ring-answer.
   1 to 60000
   2000 (default)

se.rt.<ringClass>.type
   Set the answer mode for a ringtone.
   ring
   visual
   answer
   ring-answer
Distinctive Ringtones

This feature enables you to apply a distinctive ringtone to a registered line, a specific contact, or type of call, including internal or external calls.

You can set up distinctive ringing using more than one of the following methods, however, the phone uses the highest priority method:

- Assign ringtones to specific contacts in the contact directory. This option is the first and highest in priority.
- Use the voIpProt.SIP.alertInfo.x.value and voIpProt.SIP.alertInfo.x.class parameters to map calls to specific ringtones. The value you enter depends on the call server. This option requires server support and is second in priority.
- Users can select a ringtone for each registered line on the phone from the phone menu. This option has the lowest priority.

Note: You can use the SIP alert-info header to delay the auto-answer feature. If you set delay=0 in the SIP.alert-Info header, the phone immediately auto-answers incoming calls without ringing. If you set delay=x where x=time in seconds, the phone rings for that duration of time before auto-answering incoming calls.

Distinctive Ringtone Parameters

The following list includes the parameters you can use to configure distinctive ringtones for a line, contact, or type of call.

voIpProt.SIP.alertInfo.x.class

Alert-Info fields from INVITE requests are compared to parameters as specified (x=1, 2, ..., N) and if a match is found, the behavior described in the corresponding ring class is applied.

default (default)

See the list of ring classes in Ringtone Parameters.

voIpProt.SIP.alertInfo.x.value

Specify a ringtone for a single registered line using a string to match the Alert-Info header in the incoming INVITE.

NULL (default)

reg.x.ringType

The ringer to be used for calls received by this registration. The default is the first non-silent ringer.

If you use the configuration parameters ringer13 and ringer14 on a single registered line, the phone plays SystemRing.wav.

default (default)

ringer1 to ringer24
## Ringtone Patterns

The following table lists the ring pattern names and their default descriptions.

Sampled audio files 1 to 10 all use the same built-in file unless that file has been replaced with a downloaded file.

### Ringtone Pattern Names

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Ringtone Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ringer1</td>
<td>Silent Ring</td>
<td>Silent ring</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> Silent ring provides a visual indication of an incoming call, but no audio indication.</td>
</tr>
<tr>
<td>ringer2</td>
<td>Low Trill</td>
<td>Long single A3 Db3 major warble</td>
</tr>
<tr>
<td>ringer3</td>
<td>Low Double Trill</td>
<td>Short double A3 Db3 major warble</td>
</tr>
<tr>
<td>ringer4</td>
<td>Medium Trill</td>
<td>Long single C3 E3 major warble</td>
</tr>
<tr>
<td>ringer5</td>
<td>Medium Double Trill</td>
<td>Short double C3 E3 major warble</td>
</tr>
<tr>
<td>ringer6</td>
<td>High Trill</td>
<td>Long single warble 1</td>
</tr>
<tr>
<td>ringer7</td>
<td>High Double Trill</td>
<td>Short double warble 1</td>
</tr>
<tr>
<td>ringer8</td>
<td>Highest Trill</td>
<td>Long single Gb3 A4 major warble</td>
</tr>
<tr>
<td>ringer9</td>
<td>Highest Double Trill</td>
<td>Short double Gb3 A4 major warble</td>
</tr>
<tr>
<td>ringer10</td>
<td>Beeble</td>
<td>Short double E3 major</td>
</tr>
<tr>
<td>ringer11</td>
<td>Triplet</td>
<td>Short triple C3 E3 G3 major ramp</td>
</tr>
<tr>
<td>ringer12</td>
<td>Ringback-style</td>
<td>Short double ringback</td>
</tr>
<tr>
<td>ringer13</td>
<td>Low Trill Precedence</td>
<td>Long single A3 Db3 major warble Precedence</td>
</tr>
<tr>
<td>ringer14</td>
<td>Ring Splash</td>
<td>Splash</td>
</tr>
<tr>
<td>ringer15</td>
<td>-</td>
<td>Sampled audio file 1</td>
</tr>
<tr>
<td>ringer16</td>
<td>-</td>
<td>Sampled audio file 2</td>
</tr>
<tr>
<td>ringer17</td>
<td>-</td>
<td>Sampled audio file 3</td>
</tr>
<tr>
<td>ringer18</td>
<td>-</td>
<td>Sampled audio file 4</td>
</tr>
<tr>
<td>ringer19</td>
<td>-</td>
<td>Sampled audio file 5</td>
</tr>
<tr>
<td>ringer20</td>
<td>-</td>
<td>Sampled audio file 6</td>
</tr>
</tbody>
</table>
Sound Effects
The phone uses built-in sampled audio files (SAF) in .wav format for some sound effects.
You can customize the audio sound effects that play for incoming calls and other alerts. Use synthesized tones or sampled audio files with .wav files that you download from the provisioning server or internet.
Ringtone files are stored in volatile memory which allows a maximum size of 600 KB (614400 B) for all ringtones.

Sampled Audio Files
The phones use built-in sampled audio files (SAF) in .wav file format for some sound effects.
The phones support the following sampled audio WAVE (.wav) file formats:
- mono 8 kHz G.711 u-Law - Supported on all phones
- mono G.711 (13-bit dynamic range, 8-kHz sample rate)
- G.711 A-Law - Supported on all phones
- mono L16/8000 (16-bit dynamic range, 8-kHz sample rate) - Supported on all phones
- mono 8 kHz A-law/mu-law - Supported on all phones
- L8/16000 (16-bit, 8 kHz sampling rate, mono) - Supported on all phones
- mono L16/16000 (16-bit dynamic range, 16-kHz sample rate)
- L16/16000 (16-bit, 16 kHz sampling rate, mono - Supported on all phones

Default Sample Audio Files
The following table defines the phone’s default sampled audio files.

<table>
<thead>
<tr>
<th>Default Sample Audio File Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sampled Audio File Number</strong></td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
</tbody>
</table>
**Sampled Audio File Number**

<table>
<thead>
<tr>
<th>Sampled Audio File Number</th>
<th>Default Use (Pattern Reference)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Ringer 18 (se.pat.ringer.ringer18)</td>
</tr>
<tr>
<td>5</td>
<td>Ringer 19 (se.pat.ringer.ringer19)</td>
</tr>
<tr>
<td>6</td>
<td>Ringer 20 (se.pat.ringer.ringer20)</td>
</tr>
<tr>
<td>7</td>
<td>Ringer 21 (se.pat.ringer.ringer21)</td>
</tr>
<tr>
<td>8</td>
<td>Ringer 22 (se.pat.ringer.ringer22)</td>
</tr>
<tr>
<td>9</td>
<td>Ringer 23 (se.pat.ringer.ringer23)</td>
</tr>
<tr>
<td>10</td>
<td>Ringer 24 (se.pat.ringer.ringer24)</td>
</tr>
<tr>
<td>11 to 24</td>
<td>Not Used</td>
</tr>
</tbody>
</table>

---

**Sampled Audio File Parameter**

Your custom sampled audio files must be available at the path or URL specified in the parameter `saf.x` so the phone can download the files. Make sure to include the name of the file and the .wav extension in the path.

**saf.x**

Specify a path or URL for the phone to download a custom audio file (x).

To use a Welcome sound, enable the parameter `up.welcomeSoundEnabled` and specify a file in `saf.x`. The default Welcome sound file is `Welcome.wav`.

Null (default) – The phone uses a built-in file.

Path Name – During start-up, the phone attempts to download the file at the specified path in the provisioning server.

URL – During start-up, the phone attempts to download the file from the specified URL on the Internet. Must be a RFC 1738-compliant URL to an HTTP, FTP, or TFTP wave file resource.

Note: If using TFTP, the URL must be in the following format: `tftp://<host>/[pathname]<filename>`. For example: `tftp://somehost.example.com/sounds/example.wav`.

---

**Sound Effect Patterns**

You can specify the sound effects that play for different phone functions and specify the sound effect patterns and the category.

Sound effects are defined by patterns: sequences of chord-sets, silence periods, and wave files. You can also configure sound effect patterns and ringtones. The phones use both synthesized and sampled audio sound effects.

Patterns use a simple script language that allows different chord sets or wave files to be strung together with periods of silence. The script language uses the instructions shown in the next table.
### Sound Effects Pattern Types

<table>
<thead>
<tr>
<th>Instruction</th>
<th>Meaning</th>
<th>Example</th>
</tr>
</thead>
</table>
| sampled (n) | Play sampled audio file n | se.pat.misc.SAMPLED_1.inst.1.type = "sampled" (sampled audio file instruction type)  
se.pat.misc.SAMPLED_1.inst.1.value = "2" (specifies sampled audio file 2) |
| chord (n, d) | Play chord set n (d is optional and allows the chord set ON duration to be overridden to d milliseconds) | se.pat.callProg.busyTone.inst.2.type = "chord" (chord set instruction type)  
se.pat.callProg.busyTone.inst.2.value = "busyTone" (specifies sampled audio file busyTone)  
se.pat.callProg.busyTone.inst.2.param = "2000" (override ON duration of chord set to 2000 milliseconds) |
| silence (d) | Play silence for d milliseconds (Rx audio is not muted) | se.pat.callProg.bargeIn.inst.3.type = "silence" (silence instruction type)  
se.pat.callProg.bargeIn.inst.3.value = "300" (specifies silence is to last 300 milliseconds) |
| branch (n) | Advance n instructions and execute that instruction (n must be negative and must not branch beyond the first instruction) | se.pat.callProg.alerting.inst.4.type = "branch" (branch instruction type)  
se.pat.callProg.alerting.inst.4.value = "-2" (step back 2 instructions and execute that instruction) |
| csx | Plays the tone based on the values set for Frequency, Level, OnDuration, OffDuration and Repeat time. There is a pre-defined value for each chord-set (cs). x = 1 to 12 | se.pat.misc.callParkBLFAudioNotification.inst.x.value (specify the file to play the audio tone.) |

---

### Sound Effect Pattern Parameters

There are three categories of sound effect patterns that you can use to replace cat in the parameter names: callProg (Call Progress Patterns), ringer (Ringer Patterns) and misc (Miscellaneous Patterns).

Keep the following in mind when using the parameters:
- X is the pattern name.
- Y is the instruction number.
- Both x and y need to be sequential.
- Cat is the sound effect pattern category.

**se.pat.cat.x.inst.y.type**

Sound effects name, where cat is callProg, ringer, or misc.

- sample
- chord
- silence
- branch

**se.pat.cat.x.inst.y.value**

- sampled – Sampled audio file number
- chord – Type of sound effect
- silence – Silence duration in milliseconds
- branch – Number of instructions to advance

**Call Progress Tones**

The following table lists the call progress pattern names and their descriptions.

**Call Progress Tone Pattern Names**

<table>
<thead>
<tr>
<th>Call Progress Pattern</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>alerting</td>
<td>Alerting</td>
</tr>
<tr>
<td>bargeln</td>
<td>Barge-in tone</td>
</tr>
<tr>
<td>busyTone</td>
<td>Busy tone</td>
</tr>
<tr>
<td>callWaiting</td>
<td>Call waiting tone</td>
</tr>
<tr>
<td>callWaitingLong</td>
<td>Call waiting tone long (distinctive)</td>
</tr>
<tr>
<td>confirmation</td>
<td>Confirmation tone</td>
</tr>
<tr>
<td>dialTone</td>
<td>Dial tone</td>
</tr>
<tr>
<td>howler</td>
<td>Howler tone (off-hook warning)</td>
</tr>
<tr>
<td>intercom</td>
<td>Intercom announcement tone</td>
</tr>
<tr>
<td>msgWaiting</td>
<td>Message waiting tone</td>
</tr>
<tr>
<td>precedenceCallWaiting</td>
<td>Precedence call waiting tone</td>
</tr>
<tr>
<td>Call Progress Pattern</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>precedenceRingback</td>
<td>Precedence ringback tone</td>
</tr>
<tr>
<td>preemption</td>
<td>Preemption tone</td>
</tr>
<tr>
<td>precedence</td>
<td>Precedence tone</td>
</tr>
<tr>
<td>recWarning</td>
<td>Record warning</td>
</tr>
<tr>
<td>reorder</td>
<td>Reorder tone</td>
</tr>
<tr>
<td>ringback</td>
<td>Ringback tone</td>
</tr>
<tr>
<td>secondaryDialTone</td>
<td>Secondary dial tone</td>
</tr>
<tr>
<td>stutter</td>
<td>Stuttered dial tone</td>
</tr>
</tbody>
</table>

**Miscellaneous Patterns**

The following table lists the miscellaneous patterns and their descriptions.

<table>
<thead>
<tr>
<th>Miscellaneous Pattern Names</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter Name</td>
<td>Miscellaneous Pattern Name</td>
</tr>
<tr>
<td>instantmessage</td>
<td>instant message</td>
</tr>
<tr>
<td>localHoldNotification</td>
<td>local hold notification</td>
</tr>
<tr>
<td>messageWaiting</td>
<td>message waiting</td>
</tr>
<tr>
<td>negativeConfirm</td>
<td>negative confirmation</td>
</tr>
<tr>
<td>positiveConfirm</td>
<td>positive confirmation</td>
</tr>
<tr>
<td>remoteHoldNotification</td>
<td>remote hold notification</td>
</tr>
<tr>
<td>welcome</td>
<td>welcome</td>
</tr>
<tr>
<td>callParkBLFReminderTone</td>
<td>call Park BLF Reminder Tone</td>
</tr>
<tr>
<td>callParkBLFAudioNotification</td>
<td>call Park BLF Audio Notification</td>
</tr>
</tbody>
</table>
## Supported Audio Codecs

The following table lists audio codecs supported by Poly CCX phones.

<table>
<thead>
<tr>
<th>Phone</th>
<th>Audio Codec</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>Poly CCX</td>
<td>G.711 μ -law</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>G.711a-law</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>G.722</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>G.722.1 (24kbps, 32kbps)</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>G.722.1C (48kbps)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>G.729AB</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Opus*</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>iLBC (13.33kbps, 15.2kbps)</td>
<td>0, 0</td>
</tr>
</tbody>
</table>

## Supported Audio Codec Specifications

The following table summarizes the specifications for audio codecs supported on Poly phones.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Reference</th>
<th>Raw Bit Rate</th>
<th>Maximum IP Bit Rate</th>
<th>Sample Rate</th>
<th>Default Payload Size</th>
<th>Effective Audio Bandwidth</th>
</tr>
</thead>
<tbody>
<tr>
<td>G.711 μ -law</td>
<td>RFC 1890</td>
<td>64 Kbps</td>
<td>80 Kbps</td>
<td>8 Ksps</td>
<td>20 ms</td>
<td>3.5 KHz</td>
</tr>
<tr>
<td>G.711 a-law</td>
<td>RFC 1890</td>
<td>64 Kbps</td>
<td>80 Kbps</td>
<td>8 Ksps</td>
<td>20 ms</td>
<td>3.5 KHz</td>
</tr>
<tr>
<td>G.719</td>
<td>RFC 5404</td>
<td>32 Kbps</td>
<td>48 Kbps</td>
<td>48 Ksps</td>
<td>20 ms</td>
<td>20 KHz</td>
</tr>
<tr>
<td>G.711</td>
<td>RFC 1890</td>
<td>64 Kbps</td>
<td>80 Kbps</td>
<td>16 Ksps</td>
<td>20 ms</td>
<td>7 KHz</td>
</tr>
<tr>
<td>G.722¹</td>
<td>RFC 3551</td>
<td>64 Kbps</td>
<td>80 Kbps</td>
<td>16 Ksps</td>
<td>20 ms</td>
<td>7 KHz</td>
</tr>
<tr>
<td>G.722.1</td>
<td>RFC 3047</td>
<td>24 Kbps</td>
<td>40 Kbps</td>
<td>16 Ksps</td>
<td>20 ms</td>
<td>7 KHz</td>
</tr>
<tr>
<td>Algorithm</td>
<td>Reference</td>
<td>Raw Bit Rate</td>
<td>Maximum IP Bit Rate</td>
<td>Sample Rate</td>
<td>Default Payload Size</td>
<td>Effective Audio Bandwidth</td>
</tr>
<tr>
<td>------------</td>
<td>-----------</td>
<td>--------------</td>
<td>---------------------</td>
<td>-------------</td>
<td>----------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>G.722.1C</td>
<td>G72-1C</td>
<td>24 Kbps</td>
<td>40 Kbps</td>
<td>32 Ksps</td>
<td>20 ms</td>
<td>14 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>32 Kbps</td>
<td>48 Kbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>48 Kbps</td>
<td>64 Kbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>G.729AB</td>
<td>RFC 1890</td>
<td>8 Kbps</td>
<td>24 Kbps</td>
<td>8 Ksps</td>
<td>20 ms</td>
<td>3.5 KHz</td>
</tr>
<tr>
<td>Opus</td>
<td>RFC 6716</td>
<td>8 - 24 Kbps</td>
<td>24 - 40 Kbps</td>
<td>8 Ksps</td>
<td>20 ms</td>
<td>3.5 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>16 Ksps</td>
<td></td>
<td>7 KHz</td>
</tr>
<tr>
<td>Lin16</td>
<td>RFC 1890</td>
<td>128 Kbps</td>
<td>132 Kbps</td>
<td>8 Ksps</td>
<td>10 ms</td>
<td>3.5 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>256 Kbps</td>
<td>260 Kbps</td>
<td>16 Ksps</td>
<td></td>
<td>7 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>512 Kbps</td>
<td>516 Kbps</td>
<td>32 Ksps</td>
<td></td>
<td>14 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>705.6 Kbps</td>
<td>709.6 Kbps</td>
<td>44.1 Ksps</td>
<td></td>
<td>20 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>768 Kbps</td>
<td>772 Kbps</td>
<td>48 Ksps</td>
<td></td>
<td>22 KHz</td>
</tr>
<tr>
<td>Siren 7</td>
<td>SIREN7</td>
<td>16 Kbps</td>
<td>32 Kbps</td>
<td>16 Ksps</td>
<td>20 ms</td>
<td>7 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>24 Kbps</td>
<td>40 Kbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>32 Kbps</td>
<td>48 Kbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Siren14</td>
<td>SIREN14</td>
<td>24 Kbps</td>
<td>40 Kbps</td>
<td>32 Ksps</td>
<td>20 ms</td>
<td>14 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>32 Kbps</td>
<td>48 Kbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>48 Kbps</td>
<td>64 Kbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Siren22</td>
<td>SIREN22</td>
<td>32 Kbps</td>
<td>48 Kbps</td>
<td>48 Ksps</td>
<td>20 ms</td>
<td>22 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>48 Kbps</td>
<td>64 Kbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>64 Kbps</td>
<td>80 Kbps</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>iLBC</td>
<td>RFC 3951</td>
<td>13.33 Kbps</td>
<td>31.2 Kbps</td>
<td>8 Ksps</td>
<td>30 ms</td>
<td>3.5 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td>15.2 Kbps</td>
<td>24 Kbps</td>
<td></td>
<td>20 ms</td>
<td></td>
</tr>
<tr>
<td>SILK</td>
<td>SILK</td>
<td>Skype SILK</td>
<td>6 - 20 Kbps</td>
<td>36 Kbps</td>
<td>8 Ksps</td>
<td>3.5 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>7 - 25 Kbps</td>
<td>41 Kbps</td>
<td>12 Ksps</td>
<td>5.2 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8 - 30 Kbps</td>
<td>46 Kbps</td>
<td>16 Ksps</td>
<td>7 KHz</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>12 - 40 Kbps</td>
<td>56 Kbps</td>
<td>24 Ksps</td>
<td>11 KHz</td>
</tr>
</tbody>
</table>

1 Per RFC 3551. Even though the actual sampling rate for G.722 audio is 16,000 Hz (16ksps), the RTP clock rate advertised for the G.722 payload format is 8,000 Hz because that value was erroneously assigned in RFC 1890 and must remain unchanged for backward compatibility.
Note: The network bandwidth necessary to send the encoded voice is typically 5-10% higher than the encoded bit rate due to packetization overhead. For example, a G.722.1C call at 48kbps for both the receive and transmit signals consumes about 100kbps of network bandwidth (two-way audio).

Audio Codec Parameters

You can configure a set of codec properties to improve consistency and reduce workload on the phones. Use the following parameters to specify audio codec priority on your phones.

- Permitted values to set audio codec priority are 1 - 27
- A value of 1 is the highest priority, 27 the lowest.
- If 0 or Null, the codec is disabled.
- A change to the default value does not cause a phone to restart or reboot

If a phone does not support a codec, the phone treats the value as 0, does not offer or accept calls using that codec, and continues to the codec next in priority.

voice.codecPref.G711_A
  7 (default)

voice.codecPref.G711_Mu
  6 (default)

voice.codecPref.G719.32kbps
  0 (default)

voice.codecPref.G719.48kbps
  0 (default)

voice.codecPref.G719.64kbps
  0 (default)

voice.codecPref.G722
  4 (default)

voice.codecPref.G7221.24kbps
  0 (default)

voice.codecPref.G7221_C.24kbps
  0 (default)
<table>
<thead>
<tr>
<th>Codec Preference</th>
<th>Default Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>voice.codecPref.G7221.32kbps</td>
<td>5 (default)</td>
</tr>
<tr>
<td>voice.codecPref.G7221_C.48kbps</td>
<td>2 (default)</td>
</tr>
<tr>
<td>voice.codecPref.G729_AB</td>
<td>8 (default)</td>
</tr>
<tr>
<td>voice.codecPref.iLBC.13_33kbps</td>
<td>0 (default)</td>
</tr>
<tr>
<td>voice.codecPref.iLBC.15_2kbps</td>
<td>0 (default)</td>
</tr>
<tr>
<td>voice.codecPref.Lin16.8ksps</td>
<td>0 (default)</td>
</tr>
<tr>
<td>voice.codecPref.Lin16.16ksps</td>
<td>0 (default)</td>
</tr>
<tr>
<td>voice.codecPref.Lin16.32ksps</td>
<td>0 (default)</td>
</tr>
<tr>
<td>voice.codecPref.Lin16.44_1ksps</td>
<td>0 (default)</td>
</tr>
<tr>
<td>voice.codecPref.Lin16.48ksps</td>
<td>0 (default)</td>
</tr>
<tr>
<td>voice.codecPref.Siren7.16kbps</td>
<td>0 (default)</td>
</tr>
<tr>
<td>voice.codecPref.Siren7.24kbps</td>
<td>0 (default)</td>
</tr>
<tr>
<td>voice.codecPref.Siren7.32kbps</td>
<td>0 (default)</td>
</tr>
</tbody>
</table>
voice.codecPref.Siren14.24kbps
   0 (default)
voice.codecPref.Siren14.32kbps
   0 (default)
voice.codecPref.Siren14.48kbps
   3 (default)
voice.codecPref.Siren22.32kbps
   0 (default)
voice.codecPref.Siren22.48kbps
   0 (default)
voice.codecPref.Siren22.64kbps
   1 (default)
voice.codecPref.SILK.8ksps
   0 (default)
voice.codecPref.SILK.12ksps
   0 (default)
voice.codecPref.SILK.16ksps
   0 (default)
voice.codecPref.SILK.24ksps
   0 (default)

**SILK Audio Codec Parameters**

Use the following parameters to configure the SILK audio codec.

voice.codecPref.SILK.8ksps
   Set the SILK audio codec preference for the supported codec sample rates.
   0 (default)

voice.codecPref.SILK.12ksps
   Set the SILK audio codec preference for the supported codec sample rates.
voice.codecPref.SILK.16ksps
Set the SILK audio codec preference for the supported codec sample rates.
0 (default)

voice.codecPref.SILK.24ksps
Set the SILK audio codec preference for the supported codec sample rates.
0 (default)

voice.audioProfile.SILK.8ksps.encMaxAvgBitrateKbps
Set the maximum average encoder output bitrate in kilobits per second (kbps/s) for the supported SILK sample rate.
20 kbps (default)
6 – 20 kbps

voice.audioProfile.SILK.12ksps.encMaxAvgBitrateKbps
Set the maximum average encoder output bitrate in kilobits per second (kbps/s) for the supported SILK sample rate.
25 kbps (default)
7 – 25 kbps

voice.audioProfile.SILK.16ksps.encMaxAvgBitrateKbps
Set the maximum average encoder output bitrate in kilobits per second (kbps/s) for the supported SILK sample rate.
30 kbps (default)
8 – 30 kbps

voice.audioProfile.SILK.24ksps.encMaxAvgBitrateKbps
Set the maximum average encoder output bitrate in kilobits per second (kbps/s) for the supported SILK sample rate.
40 kbps (default)
12 to 40 kbps

voice.audioProfile.SILK.encComplexity
Specify the SILK encoder complexity. The higher the number the more complex the encoding allowed.
2 (default)
0 to 2

voice.audioProfile.SILK.encDTXEnable
0 (default) – Disable Enable Discontinuous transmission (DTX).
1 - Enable DTX in the SILK encoder. Note that DTX reduces the encoder bitrate to 0bps during silence.

voice.audioProfile.SILK.encExpectedPktLossPercent
Set the SILK encoder expected network packet loss percentage.
A non-zero setting allows less inter-frame dependency to be encoded into the bitstream, resulting in increasingly larger bitrates but with an average bitrate less than that configured with voice.audioProfile.SILK.*.
0 (default)
0 to 100

voice.audioProfile.SILK.encInbandFECEnable
0 (default) - Disable inband Forward Error Correction (FEC) in the SILK encoder.
A non-zero value here causes perceptually important speech information to be sent twice: once in the normal bitstream and again at a lower bitrate in later packets, resulting in an increased bitrate.

voice.audioProfile.SILK.MaxPTime
Specify the maximum SILK packet duration in milliseconds (ms).
20 ms

voice.audioProfile.SILK.MinPTime
Specify the minimum SILK packet duration in milliseconds (ms).
20 ms

voice.audioProfile.SILK.pTime
The recommended received SILK packet duration in milliseconds (ms).
20 ms

IEEE 802.1p/Q
The phone tags all Ethernet packets it transmits with an 802.1Q VLAN header when the following occurs:
• A valid VLAN ID is specified in the phone’s network configuration.
• The phone is instructed to tag packets through Cisco Discovery Protocol (CDP) running on a connected Ethernet switch.
• A VLAN ID is obtained from DHCP or LLDP
IEEE 802.1p/Q Parameters

Use the following list to set values for IEEE 802.1p/Q parameters.

You can configure the user_priority specifically for RTP and call control packets, such as SIP signaling packets, with default settings configurable for all other packets.

The phone tags all Ethernet packets it transmits with an 802.1Q VLAN header when the following occurs:

- A valid VLAN ID specified in the phone’s network configuration.
- The phone is instructed to tag packets through Cisco Discovery Protocol (CDP) running on a connected Ethernet switch.
- A VLAN ID is obtained from DHCP or CDP.

$qos.ethernet.other.user_priority$

Set user priority for packets without a per-protocol setting.

- 2 (Default)
- 0 - 7

$qos.ethernet.rtp.video.user_priority$

Set user-priority used for Video RTP packets.

- 5 (Default)
- 0 - 7

$qos.ethernet.rtp.user_priority$

Choose the priority of voice Real-Time Protocol (RTP) packets.

- 5 (Default)
- 0 - 7

$qos.ethernet.callControl.user_priority$

Set the user-priority used for call control packets.

- 5 (Default)
- 0 - 7

Voice Quality Monitoring (VQMon)

You can configure the phones to generate various quality metrics that you can use to monitor sound and listening quality.

These metrics can be sent between the phones in RTCP XR packets, which are compliant with RFC 3611—RTP Control Extended Reports (RTCP XR). The packets are sent to a report collector as specified in draft RFC Session initiation Protocol Package for Voice Quality Reporting Event. The metrics can also be sent as SIP PUBLISH messages to a central voice quality report collector.
You can use Real Time Control Protocol Extended Report (RTCP XR) to report voice quality metrics to remote endpoints. This feature supports RFC6035 compliance as well as draft implementation for voice quality reporting.

For more information on VQMon, contact your Certified Reseller.

VQMon Reports

You can enable three types of voice quality reports:

- Alert – Generated when the call quality degrades below a configurable threshold.
- Periodic – Generated during a call at a configurable period.
- Session – Generated at the end of a call.

You can generate a wide range of performance metrics using the parameters shown in the following list. Some are based on current values, such as jitter buffer nominal delay and round trip delay, while others cover the time period from the beginning of the call until the report is sent, such as network packet loss. Some metrics are generated using other metrics as input, such as listening Mean Opinion Score (MOS), conversational MOS, listening R-factor, and conversational R-factor.

VQMon Parameters

The parameters listed in the following list configure Voice Quality Monitoring.

`voice.qualityMonitoring.collector.alert.moslq.threshold.critical`

Specify the threshold value of listening MOS score (MOS-LQ) that causes the phone to send a critical alert quality report. Configure the desired MOS value multiplied by 10.

For example, a value of 28 corresponds to the MOS score 2.8.

0 (default) - Critical alerts are not generated due to MOS-LQ.
0 - 40
Change causes system to restart or reboot.

`voice.qualityMonitoring.collector.alert.moslq.threshold.warning`

Specify the threshold value of listening MOS score (MOS-LQ) that causes phone to send a warning alert quality report. Configure the desired MOS value multiplied by 10.

For example, a configured value of 35 corresponds to the MOS score 3.5.

0 (default) - Warning alerts are not generated due to MOS-LQ.
0 - 40
Change causes system to restart or reboot.

`voice.qualityMonitoring.collector.alert.delay.threshold.critical`

Specify the threshold value of one way-delay (in milliseconds) that causes the phone to send a critical alert quality report.

One-way delay includes both network delay and end system delay.

0 (default) - Critical alerts are not generated due to one-way delay.
0 - 2000 ms
Change causes system to restart or reboot.

**voice.qualityMonitoring.collector.alert.delay.threshold.warning**
Specify the threshold value of one-way delay (in milliseconds) that causes the phone to send a critical alert quality report.
One-way delay includes both network delay and end system delay.
0 (default) - Warning alerts are not generated due to one-way delay.
0 - 2000 ms
Change causes system to restart or reboot.

**voice.qualityMonitoring.collector.enable.periodic**
0 (default) - Periodic quality reports are not generated.
1 - Periodic quality reports are generated throughout a call.
Change causes system to restart or reboot.

**voice.qualityMonitoring.collector.enable.session**
1 (default) - Reports are generated at the end of each call.
0 - Quality reports are not generated at the end of each call.
Change causes system to restart or reboot.

**voice.qualityMonitoring.collector.enable.triggeredPeriodic**
0 (default) - Alert states do not cause periodic reports to be generated.
1 - Periodic reports are generated if an alert state is critical.
2 - Period reports are generated when an alert state is either warning or critical.

**Note:** This parameter is ignored when voice.qualityMonitoring.collector.enable.periodic is 1, since reports are sent throughout the duration of a call.

Change causes system to restart or reboot.

**voice.qualityMonitoring.collector.period**
The time interval (in milliseconds) between successive periodic quality reports.
20 (default)
5 - 900 ms
Change causes system to restart or reboot.

**voice.qualityMonitoring.collector.server.x.address**
The server address of a SIP server (report collector) that accepts voice quality reports contained in SIP PUBLISH messages.
Set x to 1 as only one report collector is supported at this time.
NULL (default)
IP address or hostname
Change causes system to restart or reboot.

**voice.qualityMonitoring.collector.server.x.outboundProxy.address**

This parameter directs SIP messages related to voice quality monitoring to a separate proxy. No failover is supported for this proxy, and voice quality monitoring is not available for error scenarios.

NULL (default)
IP address or FQDN

**voice.qualityMonitoring.collector.server.x.outboundProxy.port**

Specify the port to use for the voice quality monitoring outbound proxy server.

0 (default)
0 to 65535

**voice.qualityMonitoring.collector.server.x.outboundProxy.transport**

Specify the transport protocol the phone uses to send the voice quality monitoring SIP messages.

DNSaptr (default)
TCPpreferred
UDPOnly
TLS
TCPOnly

**voice.qualityMonitoring.collector.server.x.port**

Set the port of a SIP server (report collector) that accepts voice quality reports contained in SIP PUBLISH messages.

Set x to 1 as only one report collector is supported at this time.

5060 (default)
1 to 65535

**voice.qualityMonitoring.failover.enable**

1 (default) - The phone performs a failover when voice quality SIP PUBLISH messages are unanswered by the collector server.

0 - No failover is performed; note, however, that a failover is still triggered for all other SIP messages.

This parameter is ignored if

voice.qualityMonitoring.collector.server.x.outboundProxy is enabled.
**voice.qualityMonitoring.location**

Specify the device location with a valid location string. If you do not configure a location value, you must use the default string 'Unknown'.

Unknown (default)

**voice.qualityMonitoring.rfc6035.enable**

0 (default) - The existing draft implementation is supported.
1 - Complies with RFC6035.

**voice.qualityMonitoring.rtcpxr.enable**

0 (default) - RTCP-XR packets are not generated.
1 - The packets are generated.
Change causes system to restart or reboot.
Video Features

Topics:

- Video Quality Parameters
- Video and Camera Options
- Toggling Between Audio-only or Audio-Video Calls
- Video Parameters
- I-Frames

Poly CCX 600 and 700 phones support video calls.

Note: The CCX 600 requires an optional EagleEye Mini USB camera to send video.

Video Quality Parameters

Use the following parameters to configure quality settings for video calls.

video.quality

The optimal quality for video that is sent in a call or a conference.

motion (default) — For outgoing video that has motion or movement.

sharpness — For outgoing video that has little or no movement.

Note: If you don’t select motion, moderate to heavy motion can cause the phone to drop some frames.

video.quality.content

motion (default) - For outgoing video that has motion or movement.

sharpness - For outgoing video that has little or no movement.

video.autoFullScreen

0 (default) - Video calls only use the full screen layout if it is explicitly selected by the user.

1 - Video calls use the full screen layout by default, such as when a video call is first created or when an audio call transitions to a video call.

video.callRate

The default call rate (in Kbps) to use when initially negotiating bandwidth for a video call.

512 (default) - The overlay does not time out.

128 2048
**video.forceRtcpVideoCodecControl**

0 (default) - RTCP feedback messages depend on a successful SDP negotiation of a=rtcp-fb and are not used if that negotiation is missing.

1 - The phone is forced to send RTCP feedback messages to request fast I-frame updates along with SIP INFO messages for all video calls irrespective of a successful SDP negotiation of a=rtcp-fb.

For an account of all parameter dependencies when setting I-frame requests, refer to the section I-Frames.

**video.maxCallRate**

Sets the maximum call rate that the users can select. The value set on the phone cannot exceed this value. If `video.callRate` exceeds this value, this parameter overrides `video.callRate` and this value is used as the maximum.

768 (default)

128 – 2048

---

**Video and Camera Options**

At the start of a video call, video-enabled phones, including those with a connected USB camera, transmit an RTP encapsulated video stream by default. Use the following configuration parameters to configure the video and camera options for supported cameras.

Use the global video and camera parameters to configure settings for any Poly camera.

**Video and Camera Parameters**

Use the following parameters to configure video and camera options for all Poly cameras.

**feature.fecc.enabled**

1 (default) - Enable far-end camera control.

0 - Disable far-end camera control.

Change causes system to restart or reboot.

**feature.fecc.payload**

Set the RTP payload used to receive far-end camera control data.

124 (default)

100 - 127

**homeScreen.camera.enable**

0 (default) - A **Camera** menu item is shown on the main menu.

1 - A **Camera** menu item displays on the Home Screen allowing users to pan, tilt, or zoom.
**mr.video.camera.focus.auto**

NULL (default)  
0 - Disable the camera's automatic focus.  
1 - Enable the camera's automatic focus.  
Change causes system to restart or reboot.

**mr.video.camera.focus.range**

Specify the distance to the camera's optimally-focused target.  
NULL (default)  
0  
0 - 255

**reg.x.fecc.enabled**

1 (default) - Enable far-end camera control for the line you specify with x.  
0 - Disable far-end camera control for the line.

**up.arrow.repeatDelay**

Choose the milliseconds (ms) an arrow button must be held before the arrow starts repeating in the Camera Controls menu for supported Poly USB cameras.  
500 ms (default)  
100 – 5000 ms

**up.arrow.repeatRate**

Choose the milliseconds (ms) between repeated simulated presses while an arrow button is being held down. This applies to the arrows in the Camera Controls menu for supported Poly USB cameras.  
80 ms (default)  
50 – 2000 ms

**video.camera.attendeesCountEnable**

Enable to allow camera to count attendees on a video call.  
0 (default) - Disabled  
1 - Enabled

**video.camera.autoWhiteBalance**

0 – Disable auto white balance.  
1- Enable auto white balance.

**video.camera.backlightCompensation**

NULL (default)
video.camera.brightness
Sets the brightness level of the video stream. The value range is from 0 (dimmest) to 1000 (brightest).
NULL (default)
0 - 1000

video.camera.contrast
Sets the contrast level of the video stream for all supported USB cameras. The value range is from 0 (no contrast increase) to 3 (most contrast increase), and 4 (noise reduction contrast).
NULL (default)
0 - 1000

video.camera.controlStyle
Choose whether to control pan and tilt Poly USB cameras with directional arrow buttons or separate pan/tilt sliders.
Default (default)
Alternate

video.camera.flickerAvoidance
Sets the flicker avoidance for all supported USB cameras.
Null (default) - Flicker avoidance is automatic.
50hz AC power frequency flicker avoidance (Europe/Asia).
60hz AC power frequency flicker avoidance (North America).
Disabled

video.camera.focus.auto
NULL (default)
0 - Disable the camera's automatic focus.
1 - Enable the camera's automatic focus.
Change causes system to restart or reboot.

video.camera.focus.range
Specify the distance to the camera's optimally-focused target.
NULL (default)
0 - 255
**video.camera.frameRate**

Sets the target frame rate (frames per second) for all supported USB cameras. Values indicate a fixed frame rate from 5 (least smooth) to 30 (most smooth).

- 25 (default)
- 5 - 30

If `video.camera.frameRate` is set to a decimal number, the value 25 is used instead.

**video.camera.menuLocation**

Specify if camera settings, including preset storage and modifications, display under the **Advanced** menu for administrators or the **Basic** menu for users.

- Basic (default)
- Advanced

**video.camera.preset.home.pan**

Set the pan coordinate for a camera home preset.

Default values are set by and depend on the camera you are using.

- 0 – 1000

**video.camera.preset.home.tilt**

Set the tilt coordinate for a camera home preset.

Default values are set by and depend on the camera you are using.

- 0 - 1000

**video.camera.preset.home.uponIdle.delay**

Set the number of minutes after the idle timeout expires to move the camera to the home preset.

- 0 (default)
- 0 - 3600

**video.camera.preset.home.uponIdle.enabled**

0 (default) – Do not move the camera to the home preset when the camera is idle.

- 1 - Move the camera to the home preset when the camera is idle.

**video.camera.preset.home.zoom**

Set the zoom coordinate for a camera home preset.

Default values are set by and depend on the camera you are using.

- 0 - 1000

**video.camera.presetIndex**

Set the number of presets available.
(default)  
0 - 1000

**video.camera.saturation**  
Sets the saturation level of video captured by any supported USB camera.  
NULL (default)  
0 - 1000

**video.camera.sharpness**  
Sets the sharpness level of video captured.  
NULL (default)  
0 - 1000

**video.camera.whiteBalance**  
Use to correct the white balance tint of video captured by any supported USB camera.  
NULL (default)  
0 - 1000

**video.localCameraView.callState**  
Set to determine how the local camera view (LCV) displays when the phone is in a call.  
1 (default) - The local camera view displays on the connected monitor.  
0 - The local camera view does not display on the connected monitor.  
This parameter applies only when **video.localCameraView.userControl** is set to PerSession or Hidden.

**video.localCameraView.idleState**  
Set to determine how the local call view (LCV) displays when the phone is idle.  
0 (default) - Does not display the LCV when the phone is idle.  
1 - Displays the LCV in picture-in-picture (PIP) when the phone is idle.  
This parameter applies only when **video.localCameraView.userControl** is set to PerSession or Hidden.

**video.localCameraView.fullScreen.callState**  
Set to determine how the local call view (LCV) displays in full screen when the phone is in a call.  
0 (default) - Displays the LCV in PIP during a call.  
1 - Displays the LCV in full screen during a call.

**video.localCameraView.fullScreen.idleState**  
Set to determine how the local call view (LCV) displays in full screen when the phone is idle.
0 (default) - Does not display the LCV when the phone is idle.
1 - Displays the LCV in picture-in-picture (PIP) when the phone is idle.

**video.localCameraView.fullScreen.userControl**
Set to enable users to control how the local call view (LCV) displays during a call.
Persistent (default) - Enables users to access the Layout menu and control how the LCV displays before a call.
PerSession - Enables users to access the Layout menu and control how the LCV displays before or during a call.
Hidden - Hides the Layout menu so that users cannot control how the LCV displays.

**video.vc4Decode.overrunTolerance**
Set the overrun errors per second for video decoder tolerance. If the decoder generates more overrun errors than the number you set, the CCX drops SVC video layer 1 to reduce the decoder load.
0 (default) – Disable tolerance for decoder overrun errors.
0 – 100 overrun errors per second

**Toggling Between Audio-only or Audio-Video Calls**
You can enable users to toggle between audio-only and audio-video calls.
When this feature is enabled on a phone using video capabilities, you can toggle calls between audio-only or audio-video.
This feature applies only to outbound calls from your phone; incoming video calls to your phone are answered using video even when you set the feature to use audio-only.
When the phone is registered, you can do the following:
- Use `video.callMode.default` to begin calls as audio-video or audio only. By default, calls begin as audio-video. After a video call has ended, the phone returns to audio-only.
  - If you set this parameter to audio, users can choose to add Video to the call.
- Use `up.homeScreen.audioCall.enabled` to enable a Home screen icon that allows users to make audio-only calls. Far-end users can add video during a call if the far-end device is video capable.

**Audio-only or Audio-Video Call Parameters**
The following parameters configure whether the phone starts a call with audio and video.

**up.homeScreen.audioCall.enabled**
0 (default) - Disable a Home screen icon that allows users to make audio-only calls.
1 - Enable a Home screen icon that allows users to make audio-only calls.
Devices that support video calling show an 'Audio Call' button on the Home screen to initiate audio-only calls.
_video.autoStartVideoTx_

1 (default) - Automatically begin video to the far side when you start a call.
0 - Video to the far side does not begin.

_audioVideoToggle.callMode.persistent_

0 - Resets the call mode set by a user to the default.
1 (default) - Maintains the call mode set by a user.

_audioVideoToggle.callMode.persistent_

0 - Resets the call mode set by a user to the default.
1 - Maintains the call mode set by a user.

_video.callMode.default_

Allow the user to begin calls as audio-only or with video.

_video (default)_

_audio - Set the initial call to audio only and video may be added during a call._

## Video Parameters

Use the following parameters to configure video features on video-capable phones.

_video.allowWithSource_

Restricts sending video codec negotiation in Session Description Protocol (SDP).

0 (default)
0 or 1

_video.autoFullScreen_

0 (default) - Video calls use the full screen layout, only if explicitly selected by the user.
1 - Video calls use the full screen layout by default.

_video.conf.profile_

Sets the video resolution to large window in all layouts.
540p (default)
1080p
720p
360p
240p
180p
**video.dynamicControlMethod**

0 (default)

1 - The first I-Frame request uses the method defined by `video.forceRtcpVideoCodecControl` and subsequent requests alternate between RTCP-FB and SIP INFO.

To set other methods for I-frame requests, refer the parameter `video.forceRtcpVideoCodecControl`.

**video.iFrame.delay**

0 (default)

1 - 10 seconds - Transmits an extra I-frame after the video starts.

You can configure the amount of delay from the start of video until the I-frame is sent up to 10 seconds.

Change causes system to restart or reboot.

**video.iFrame.minPeriod**

Time taken before sending a second I-frame in response to requests from the far end.

2 (default)

1 - 60

**video.iFrame.onPacketLoss**

0 (default)

1 - Transmits an I-frame to the far end when video RTP packet loss occurs.

**video.iFrame.period.onBoard**

Set the I-Frame interval used for the VC4 encoder.

180 (default)

300 maximum

**Video Codec Preference Parameters**

Use the following video codec parameters to specify video codec preferences.

To disable codecs, set the value to 0. A value of 1 indicates the codec is the most preferred and has highest priority.

**video.codecPref.H261**

Sets the H.261 payload type.

6 (default)

0 - 8
video.codecPref.H264
Sets the H.264 payload type.
4 (default)
0 - 8

video.codecPref.H263 1998
Sets the H.263 payload type.
5 (default)
0 - 8

video.codecPref.H263
5 (default)
0 - 8

video.codecPref.H264
4 (default)
0 - 8

video.codecPref.H264.packetizationMode0
Sets the H.264 payload type when packetization mode is set to 0.
5 (default)
0 - 8

video.codecPref.H264HP
Sets the H.264 High Profile video codec preference priority.
2 (default)
0 - 8

video.codecPref.H264HP.packetizationMode0
Sets the H.264 high profile payload type when packetization mode is set to 0.
3 (default)
0 - 8

video.codecPref.H264SVC

video.codecPref.Xdata
Sets the Remote Desktop Protocol (RDP) codec preference priority.
7 (default)
0 - 8
1 - Codec has highest priority.

**video.codecPref.XH264UC**
Sets the Microsoft H.264 UC video codec preference priority.
1 (default)
0 - 8

**video.codecPref.XUlpFecUC**
Sets the forward error correction (FEC) codec priority.
8 (default)
0 - 8

## I-Frames

When video streams initialize, devices transmit video packets called I-frames (reference frames) that contain information to display a complete picture.

The devices subsequently send smaller and less complete frames, known as P-frames, to consume less bandwidth. Due to packet loss, jitter, or corruption, devices occasionally need to make multiple requests for a complete I-frame in order to reset the full frame, after which devices can revert to P-frame updates.

You can set parameters to control an I-frame request. The following table indicates parameter dependencies and messaging behavior when setting an I-frame request method.

### I-Frame Parameter Dependencies

<table>
<thead>
<tr>
<th>video.forceRtcpVideoCodecControl</th>
<th>video.dynamicControlMethod</th>
<th>voIpProt.SDP.offer.rtcpVideoCodecControl</th>
<th>Behavior when requesting video I-frame updates</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0 (n/a)</td>
<td>0</td>
<td>Only SIP INFO messages are sent. No RTCP-FB is offered in SDP.</td>
</tr>
<tr>
<td>0</td>
<td>1 (n/a)</td>
<td>0</td>
<td>Only SIP INFO messages are sent. No RTCP-FB is offered in SDP.</td>
</tr>
<tr>
<td>0</td>
<td>0 (n/a)</td>
<td>1</td>
<td>RTCP-FB is offered in SDP. If SDP responses do not contain the required RTCP-FB attribute, then only SIP INFO requests are used.</td>
</tr>
<tr>
<td>0</td>
<td>1 (N/A)</td>
<td>1</td>
<td>RTCP-FB is offered in SDP. If SDP responses do not contain the required RTCP-FB attribute, then only SIP INFO requests are used.</td>
</tr>
<tr>
<td>video.forceRtcpVideoCodecControl</td>
<td>video.dynamicControlMethod</td>
<td>voIPProt.SDP.offer.rtcpVideoCodecControl</td>
<td>Behavior when requesting video I-frame updates</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>---------------------------</td>
<td>------------------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>The SDP attribute a=rtcp-fb is not included in SDP offers. Both RTCP-FB and SIP INFO messages are attempted.</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>The SDP attribute a=rtcp-fb is not included in SDP offers. Both RTCP-FB and SIP INFO messages are attempted. If no RTCP-FB messages are received, only SIP INFO messages are sent. If no response is received for SIP INFO messages then, again, both RTCP-FB and SIP INFO messages are attempted.</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>RTCP-FB is offered in SDP. Even if the SDP response does not include an accepted a=rtcp-fb attribute both RTCP-FB and SIP INFO messages are sent.</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>RTCP-FB is offered in SDP. Even if the SDP response does not include an accepted a=rtcp-fb attribute both RTCP-FB and SIP INFO messages are sent initially. If no RTCP-FB response is received, only SIP INFO messages are sent afterwards.</td>
</tr>
</tbody>
</table>
Phone Display Features

Topics:

• Time Zone Location Description
• Time and Date
• Custom User Interface Parameters
• Default Phone Screen
• Graphic Display Background
• Digital Picture Frame
• Background Image Lock
• Phone Languages
• Pinyin Text Input
• Hide the MAC Address
• Digital Phone Label
• Unique Line Labels for Registration Lines
• LED Indicators
• Capture Your Phone's Screen
• Line View Pages

This section explains features you can configure for the phone’s screen display and lists parameters you can use to configure these features.

Time Zone Location Description

There are two parameters that configure a time zone location description for their associated GMT offset.

• device.sntp.gmtOffsetcityID If you are not provisioning phones manually from the phone menu or Web Configuration Utility and you are setting the device.sntp.gmtOffset parameter, then you must configure device.sntp.gmtOffsetcityID to ensure that the correct time zone location description displays on the phone menu and Web Configuration Utility. The time zone location description is set automatically if you set the device.sntp.gmtOffset parameter manually using the phone menu or Web Configuration Utility.

• tcpIpApp.sntp.gmtOffsetcityID If you are not provisioning phones manually from the Web Configuration Utility and you are setting the tcpIpApp.sntp.gmtOffset parameter, then you must configure tcpIpApp.sntp.gmtOffsetcityID to ensure that the correct time zone location description displays on the Web Configuration Utility. The time zone location description is set automatically if you set the tcpIpApp.sntp.gmtOffset parameter manually using the Web Configuration Utility.
### Time Zone Location Parameters

The following parameters configure time zone location.

#### Time Zone Location Parameter Values

<table>
<thead>
<tr>
<th>Permitted Value</th>
<th>Time Zone Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(GMT -12:00) Eniwetok, Kwajalein</td>
</tr>
<tr>
<td>1</td>
<td>(GMT -11:00) Midway Island</td>
</tr>
<tr>
<td>2</td>
<td>(GMT -10:00) Hawaii</td>
</tr>
<tr>
<td>3</td>
<td>(GMT -9:00) Alaska</td>
</tr>
<tr>
<td>4</td>
<td>(GMT -8:00) Pacific Time (US &amp; Canada)</td>
</tr>
<tr>
<td>5</td>
<td>(GMT -8:00) Baja California</td>
</tr>
<tr>
<td>6</td>
<td>(GMT -7:00) Mountain Time (US &amp; Canada)</td>
</tr>
<tr>
<td>7</td>
<td>(GMT -7:00) Chihuahua, La Paz</td>
</tr>
<tr>
<td>8</td>
<td>(GMT -7:00) Mazatlan</td>
</tr>
<tr>
<td>9</td>
<td>(GMT -7:00) Arizona</td>
</tr>
<tr>
<td>10</td>
<td>(GMT -6:00) Central Time (US &amp; Canada)</td>
</tr>
<tr>
<td>11</td>
<td>(GMT -6:00) Mexico City</td>
</tr>
<tr>
<td>12</td>
<td>(GMT -6:00) Saskatchewan</td>
</tr>
<tr>
<td>13</td>
<td>(GMT -6:00) Guadalajara</td>
</tr>
<tr>
<td>14</td>
<td>(GMT -6:00) Monterrey</td>
</tr>
<tr>
<td>15</td>
<td>(GMT -6:00) Central America</td>
</tr>
<tr>
<td>16</td>
<td>(GMT -5:00) Eastern Time (US &amp; Canada)</td>
</tr>
<tr>
<td>17</td>
<td>(GMT -5:00) Indiana (East)</td>
</tr>
<tr>
<td>18</td>
<td>(GMT -5:00) Bogota, Lima</td>
</tr>
<tr>
<td>19</td>
<td>(GMT -5:00) Quito</td>
</tr>
<tr>
<td>20</td>
<td>(GMT -4:30) Caracas</td>
</tr>
<tr>
<td>21</td>
<td>(GMT -4:00) Atlantic Time (Canada)</td>
</tr>
<tr>
<td>22</td>
<td>(GMT -4:00) San Juan</td>
</tr>
<tr>
<td>23</td>
<td>(GMT -4:00) Manaus, La Paz</td>
</tr>
<tr>
<td>24</td>
<td>(GMT -4:00) Asuncion, Cuiaba</td>
</tr>
<tr>
<td>25</td>
<td>(GMT -4:00) Georgetown</td>
</tr>
<tr>
<td>26</td>
<td>(GMT -3:30) Newfoundland</td>
</tr>
<tr>
<td>27</td>
<td>(GMT -3:00) Brasilia</td>
</tr>
<tr>
<td>28</td>
<td>(GMT -3:00) Buenos Aires</td>
</tr>
<tr>
<td>29</td>
<td>(GMT -3:00) Greenland</td>
</tr>
<tr>
<td>30</td>
<td>(GMT -3:00) Cayenne, Fortaleza</td>
</tr>
<tr>
<td>Permitted Value</td>
<td>Time Zone Description</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>31</td>
<td>(GMT -3:00) Montevideo</td>
</tr>
<tr>
<td>32</td>
<td>(GMT -3:00) Salvador</td>
</tr>
<tr>
<td>33</td>
<td>(GMT -3:00) Santiago</td>
</tr>
<tr>
<td>34</td>
<td>(GMT -2:00) Mid-Atlantic</td>
</tr>
<tr>
<td>35</td>
<td>(GMT -1:00) Azores</td>
</tr>
<tr>
<td>36</td>
<td>(GMT -1:00) Cape Verde Islands</td>
</tr>
<tr>
<td>37</td>
<td>(GMT 0:00) Western Europe Time</td>
</tr>
<tr>
<td>38</td>
<td>(GMT 0:00) London, Lisbon</td>
</tr>
<tr>
<td>39</td>
<td>(GMT 0:00) Casablanca</td>
</tr>
<tr>
<td>40</td>
<td>(GMT 0:00) Dublin</td>
</tr>
<tr>
<td>41</td>
<td>(GMT 0:00) Edinburgh</td>
</tr>
<tr>
<td>42</td>
<td>(GMT 0:00) Monrovia</td>
</tr>
<tr>
<td>43</td>
<td>(GMT 0:00) Reykjavik</td>
</tr>
<tr>
<td>44</td>
<td>(GMT +1:00) Belgrade</td>
</tr>
<tr>
<td>45</td>
<td>(GMT +1:00) Bratislava</td>
</tr>
<tr>
<td>46</td>
<td>(GMT +1:00) Budapest</td>
</tr>
<tr>
<td>47</td>
<td>(GMT +1:00) Ljubljana</td>
</tr>
<tr>
<td>48</td>
<td>(GMT +1:00) Prague</td>
</tr>
<tr>
<td>49</td>
<td>(GMT +1:00) Sarajevo, Skopje</td>
</tr>
<tr>
<td>50</td>
<td>(GMT +1:00) Warsaw, Zagreb</td>
</tr>
<tr>
<td>51</td>
<td>GMT +1:00 Brussels</td>
</tr>
<tr>
<td>52</td>
<td>GMT +1:00 Copenhagen</td>
</tr>
<tr>
<td>53</td>
<td>GMT +1:00 Madrid, Paris</td>
</tr>
<tr>
<td>54</td>
<td>GMT +1:00 Amsterdam, Berlin</td>
</tr>
<tr>
<td>55</td>
<td>GMT +1:00 Bern, Rome</td>
</tr>
<tr>
<td>56</td>
<td>GMT +1:00 Stockholm, Vienna</td>
</tr>
<tr>
<td>57</td>
<td>GMT +1:00 West Central Africa</td>
</tr>
<tr>
<td>58</td>
<td>GMT +1:00 Windhoek</td>
</tr>
<tr>
<td>59</td>
<td>GMT +2:00 Bucharest, Cairo</td>
</tr>
<tr>
<td>60</td>
<td>GMT +2:00 Amman, Beirut</td>
</tr>
<tr>
<td>Permitted Value</td>
<td>Time Zone Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>61</td>
<td>(GMT +2:00) Helsinki,Kyiv</td>
</tr>
<tr>
<td>62</td>
<td>(GMT +2:00) Riga,Sofia</td>
</tr>
<tr>
<td>63</td>
<td>(GMT +2:00) Tallinn,Vilnius</td>
</tr>
<tr>
<td>64</td>
<td>(GMT +2:00) Athens,Istanbul</td>
</tr>
<tr>
<td>65</td>
<td>(GMT +2:00) Damascus</td>
</tr>
<tr>
<td>66</td>
<td>(GMT +2:00) E.Europe</td>
</tr>
<tr>
<td>67</td>
<td>(GMT +2:00) Harare,Pretoria</td>
</tr>
<tr>
<td>68</td>
<td>(GMT +2:00) Jerusalem</td>
</tr>
<tr>
<td>69</td>
<td>(GMT +2:00) Kaliningrad (RTZ 1)</td>
</tr>
<tr>
<td>70</td>
<td>(GMT +2:00) Tripoli</td>
</tr>
<tr>
<td>71</td>
<td>(GMT +3:00) Moscow</td>
</tr>
<tr>
<td>72</td>
<td>(GMT +3:00) St.Petersburg</td>
</tr>
<tr>
<td>73</td>
<td>(GMT +3:00) Volgograd (RTZ 2)</td>
</tr>
<tr>
<td>74</td>
<td>(GMT +3:00) Kuwait,Riyadh</td>
</tr>
<tr>
<td>75</td>
<td>(GMT +3:00) Nairobi</td>
</tr>
<tr>
<td>76</td>
<td>(GMT +3:00) Baghdad</td>
</tr>
<tr>
<td>77</td>
<td>(GMT +3:30) Tehran</td>
</tr>
<tr>
<td>79</td>
<td>(GMT +4:00) Abu Dhabi,Muscat</td>
</tr>
<tr>
<td>80</td>
<td>(GMT +4:00) Baku,Tbilisi</td>
</tr>
<tr>
<td>81</td>
<td>(GMT +4:00) Izhevsk,Samara (RTZ 3)</td>
</tr>
<tr>
<td>82</td>
<td>(GMT +4:00) Port Louis</td>
</tr>
<tr>
<td>83</td>
<td>(GMT +4:00) Yerevan</td>
</tr>
<tr>
<td>84</td>
<td>(GMT +4:30) Kabul</td>
</tr>
<tr>
<td>85</td>
<td>(GMT +5:00) Yekaterinburg (RTZ 4)</td>
</tr>
<tr>
<td>86</td>
<td>(GMT +5:00) Islamabad</td>
</tr>
<tr>
<td>87</td>
<td>(GMT +5:00) Karachi</td>
</tr>
<tr>
<td>88</td>
<td>(GMT +5:00) Tashkent</td>
</tr>
<tr>
<td>89</td>
<td>(GMT +5:30) Mumbai,Chennai</td>
</tr>
<tr>
<td>90</td>
<td>(GMT +5:30) Kolkata,New Delhi</td>
</tr>
<tr>
<td>Permitted Value</td>
<td>Time Zone Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------------------------------------------------</td>
</tr>
<tr>
<td>91</td>
<td>(GMT +5:30) Sri Jayawardenepura</td>
</tr>
<tr>
<td>92</td>
<td>(GMT +5:45) Kathmandu</td>
</tr>
<tr>
<td>93</td>
<td>(GMT +6:00) Astana,Dhaka</td>
</tr>
<tr>
<td>94</td>
<td>(GMT +6:00) Almaty</td>
</tr>
<tr>
<td>95</td>
<td>(GMT +6:00) Novosibirsk (RTZ 5)</td>
</tr>
<tr>
<td>96</td>
<td>(GMT +6:30) Yangon (Rangoon)</td>
</tr>
<tr>
<td>97</td>
<td>(GMT +7:00) Bangkok,Hanoi</td>
</tr>
<tr>
<td>98</td>
<td>(GMT +7:00) Jakarta</td>
</tr>
<tr>
<td>99</td>
<td>(GMT +7:00) Krasnoyarsk (RTZ 6)</td>
</tr>
<tr>
<td>100</td>
<td>(GMT +8:00) Beijing,Chongqing</td>
</tr>
<tr>
<td>101</td>
<td>(GMT +8:00) Hong Kong,Urumqi</td>
</tr>
<tr>
<td>102</td>
<td>(GMT +8:00) Kuala Lumpur</td>
</tr>
<tr>
<td>103</td>
<td>(GMT +8:00) Singapore</td>
</tr>
<tr>
<td>104</td>
<td>(GMT +8:00) Taipei,Perth</td>
</tr>
<tr>
<td>105</td>
<td>(GMT +8:00) Irkutsk (RTZ 7)</td>
</tr>
<tr>
<td>106</td>
<td>(GMT +8:00) Ulaanbaatar</td>
</tr>
<tr>
<td>107</td>
<td>(GMT +9:00) Tokyo,Seoul,Osaka</td>
</tr>
<tr>
<td>108</td>
<td>(GMT +9:00) Sapporo,Yakutsk (RTZ 8)</td>
</tr>
<tr>
<td>109</td>
<td>(GMT +9:30) Adelaide,Darwin</td>
</tr>
<tr>
<td>110</td>
<td>(GMT +10:00) Canberra</td>
</tr>
<tr>
<td>111</td>
<td>(GMT +10:00) Magadan (RTZ 9)</td>
</tr>
<tr>
<td>112</td>
<td>(GMT +10:00) Melbourne</td>
</tr>
<tr>
<td>113</td>
<td>(GMT +10:00) Sydney,Brisbane</td>
</tr>
<tr>
<td>114</td>
<td>(GMT +10:00) Hobart</td>
</tr>
<tr>
<td>115</td>
<td>(GMT +10:00) Vladivostok</td>
</tr>
<tr>
<td>116</td>
<td>(GMT +10:00) Guam,Port Moresby</td>
</tr>
<tr>
<td>117</td>
<td>(GMT +11:00) Solomon Islands</td>
</tr>
<tr>
<td>118</td>
<td>(GMT +11:00) New Caledonia</td>
</tr>
<tr>
<td>119</td>
<td>(GMT +11:00) Chokurdakh (RTZ 10)</td>
</tr>
<tr>
<td>120</td>
<td>(GMT +12:00) Fiji Islands</td>
</tr>
<tr>
<td>Permitted Value</td>
<td>Time Zone Description</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>121</td>
<td>(GMT +12:00) Auckland, Anadyr</td>
</tr>
<tr>
<td>122</td>
<td>(GMT +12:00) Petropavlovsk-Kamchatsky (RTZ 11)</td>
</tr>
<tr>
<td>123</td>
<td>(GMT +12:00) Wellington</td>
</tr>
<tr>
<td>124</td>
<td>(GMT +12:00) Marshall Islands</td>
</tr>
<tr>
<td>125</td>
<td>(GMT +13:00) Nuku'alofa</td>
</tr>
<tr>
<td>126</td>
<td>(GMT +13:00) Samoa</td>
</tr>
</tbody>
</table>

**Time and Date**

A clock and calendar display on the phones by default.

You can choose how to display the time and date for your time zone in several formats, or you can disable the display of the time and date. You can also set the time and date format to display differently when the phone is in certain modes. For example, the display format can change when the phone goes from idle mode to an active call.

To have the most accurate time, you have to synchronize the phone to the Simple Network Time Protocol (SNTP) time server. Until a successful SNTP response is received, the phone continuously flashes the time and date to indicate that they are not accurate.

The time and date display on the phones in PSTN mode and are set by an incoming call with a supported caller ID standard, or when the phone is connected to Ethernet and you enable the date and time display.

**Time and Date Display Parameters**

Use the parameters in the following list to configure time and display options.

`up.localClockEnabled`

- Specifies whether or not the date and time are shown on the idle display.
  - 1 (Default) - Date and time are shown.
  - 0 - Date and time are hidden.

`lcl.datetime.date.dateTop`

- 1 - Displays the date above time.
  - 0 (default) - Displays the time above date.

`lcl.datetime.date.format`

- The phone displays day and date. The field may contain 0, 1, or 2 commas which can occur only between characters and only one at a time.
  - For example: D,dM = Thursday, 3 July or Md,D = July 3, Thursday.
  - "D,dM" (default)
  - String
**lcl.datetime.date.longFormat**

1 (default) - Displays the day and month in long format (Friday/November).
0 - Displays the day and month in abbreviated format (Fri/Nov).

**lcl.datetime.time.24HourClock**

1 (default) - Displays the time in 24-hour clock mode.
0 - Displays the time in 12-hour clock mode.

**tcpIpApp.sntp.address**

Specifies the SNTP server address.

NULL (default)
Valid hostname or IP address.

**tcpIpApp.sntp.AQuery**

Specifies a query to return hostnames.

0 (default) - Queries to resolve the SNTP hostname are performed using DNS SRV.
1 - Query the hostname for a DNS A record.

**tcpIpApp.sntp.address.overrideDHCP**

0 (Default) - DHCP values for the SNTP server address are used.
1 - SNTP parameters override the DHCP values.

**tcpIpApp.sntp.daylightSavings.enable**

Enable or disable Daylight Savings Time rules to the displayed time.

1 (Default) - Enabled
0 - Disabled

**tcpIpApp.sntp.daylightSavings.fixedDayEnable**

0 (Default) - Month, date, and dayOfWeek are used in the DST calculation.
1 - Only month and date are used in the DST calculation.

**tcpIpApp.sntp.daylightSavings.start.date**

Start date for daylight savings time. Range is 1 to 31.

8 (Default) - Second occurrence in the month after DST starts.
0 - If fixedDayEnable is set to 0, this value specifies the occurrence of dayOfWeek when DST should start.
1 - If fixedDayEnable is set to 1, this value is the day of the month to start DST.
15 - Third occurrence.
22 - Fourth occurrence.
Example: If value is set to 15, DST starts on the third dayOfWeek of the month.

tcpIpApp.sntp.daylightSavings.start.dayOfWeek

Specifies the day of the week to start DST. This parameter is not used if fixedDayEnable is set to 1.

1 (Default) - Sunday
1-7 where the integer entered corresponds to a day of the week. For example, 1 = Sunday, 2 = Monday, and so on to 7 = Saturday.

tcpIpApp.sntp.daylightSavings.start.dayOfWeek.lastInMonth

0 (Default)

1 - DST starts on the last dayOfWeek of the month and the start.date is ignored.

Note: This parameter is not used if fixedDayEnable is set to 1.

tcpIpApp.sntp.daylightSavings.start.month

Specifies the month to start DST.

3 (Default) - March

1-12 where the integer entered corresponds to a month of the year. For example, 1 = January, 2 = February and so on to 12 = December.

tcpIpApp.sntp.daylightSavings.start.time

Specifies the time of day to start DST in 24-hour clock format. Range is 0 to 23.

2 (Default) - 2 a.m.

0 - 23 where the integer entered corresponds to the hour on in a 24 span. For example, 0 = 12 AM, 1 = 1 AM, and so on to 23 = 11 PM.

tcpIpApp.sntp.daylightSavings.stop.date

Specifies the stop date for daylight savings time. Range is 1 to 31.

1 (Default) - If fixedDayEnable is set to 1, the value of this parameter is the day of the month to stop DST. Set 1 for the first occurrence in the month.

0 - If fixedDayEnable is set to 0, this value specifies the dayOfWeek when DST should stop.

8 - Second occurrence.
15 - Third occurrence.
22 - Fourth occurrence.

Example: If set to 22, DST stops on the fourth dayOfWeek in the month.

tcpIpApp.sntp.daylightSavings.stop.dayOfWeek

Day of the week to stop DST.
1 (default) - Sunday
1-7 where the integer entered corresponds to a day of the week. For example, 1 = Sunday, 2 = Monday, and so on to 7 = Saturday.

**Note:** Parameter is not used if `fixedDayEnable` is set to 1.

tcpIpApp.sntp.daylightSavings.stop.dayOfWeek.lastInMonth
1 - DST stops on the last `dayOfWeek` of the month and the `stop.date` is ignored.
Parameter is not used if `fixedDayEnable` is set to 1.

tcpIpApp.sntp.daylightSavings.stop.month
Specifies the month to stop DST. Range is 1 to 12.
11 (Default) - November
1-12 where the integer entered corresponds to a month of the year. For example, 1 = January, 2 = February and so on to 12 = December.

tcpIpApp.sntp.daylightSavings.stop.time
Specifies the time of day to stop DST in 24-hour clock format. Range is 0 to 23.
2 (Default) - 2 a.m.
0 - 23 where the integer entered corresponds to the hour on in a 24 span. For example, 0 = 12 AM, 1 = 1 AM, and so on to 23 = 11 PM.

tcpIpApp.sntp.gmtOffset
Specifies the offset in seconds of the local time zone from GMT.
0 (Default) - GMT
3600 seconds = 1 hour
-3600 seconds = -1 hour
Positive or negative integer

tcpIpApp.sntp.gmtOffset.cityID
You must disable `tcpIpApp.sntp.daylightSavings.enable` for the phone to display daylight savings time according to `gmtOffset.cityID`.
NULL (Default)
For descriptions of all values, refer to Time Zone Location Description.
0 - 127

tcpIpApp.sntp.gmtOffset.overrideDHCP
0 (Default) - The DHCP values for the GMT offset are used.
1 - The SNTP values for the GMT offset are used.
**tcpIpApp.sntp.resyncPeriod**

Specifies the period of time (in seconds) that passes before the phone resynchronizes with the SNTP server.

86400 (Default). 86400 seconds is 24 hours.

Positive integer

**tcpIpApp.sntp.retryDnsPeriod**

Sets a retry period for DNS queries. The DNS retry period is affected by other DNS queries made on the phone. If the phone makes a query for another service during the retry period, such as SIP registration, and receives no response, the Network Time Protocol (NTP) DNS query is omitted to limit the retry attempts to the unresponsive server. If no other DNS attempts are made by other services, the retry period is not affected. If the DNS server becomes responsive to another service, NTP immediately retries the DNS query.

86400 (Default). 86400 seconds is 24 hours.

60 - 2147483647 seconds

**Date Formats**

Use the following table to choose values for the lcl.datetime.date.format and lcl.datetime.date.longformat parameters. The table shows values for Friday, August 19, 2011 as an example.

### Date Formats

<table>
<thead>
<tr>
<th>lcl.datetime.date.format</th>
<th>lcl.datetime.date.longformat</th>
<th>Date Displayed on Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>dM,D</td>
<td>0</td>
<td>19 Aug, Fri</td>
</tr>
<tr>
<td>dM,D</td>
<td>1</td>
<td>19 August, Friday</td>
</tr>
<tr>
<td>Md,D</td>
<td>0</td>
<td>Aug 19, Fri</td>
</tr>
<tr>
<td>Md,D</td>
<td>1</td>
<td>August 19, Friday</td>
</tr>
<tr>
<td>D,dM</td>
<td>0</td>
<td>Fri, 19 Aug</td>
</tr>
<tr>
<td>D,dM</td>
<td>1</td>
<td>Friday, August 19</td>
</tr>
<tr>
<td>DD/MM/YY</td>
<td>n/a</td>
<td>19/08/11</td>
</tr>
<tr>
<td>DD/MM/YYYY</td>
<td>n/a</td>
<td>19/08/2011</td>
</tr>
<tr>
<td>MM/DD/YY</td>
<td>n/a</td>
<td>08/19/11</td>
</tr>
<tr>
<td>MM/DD/YYYY</td>
<td>n/a</td>
<td>08/19/2011</td>
</tr>
<tr>
<td>YY/MM/DD</td>
<td>n/a</td>
<td>11/08/19</td>
</tr>
<tr>
<td>YYYY/MM/DD</td>
<td>n/a</td>
<td>2011/08/11</td>
</tr>
</tbody>
</table>
Custom User Interface Parameters

You can configure custom colors for the user interface using the parameters in the following list.

**ui.home.background**
- Set the color of the background of the Home screen.
- Null (default)
- #RRGGBB color codes
- Change causes system to restart or reboot.

**ui.menu.background**
- Set the background color of the Menu screen.
- Null (default)
- #RRGGBB color codes
- Change causes system to restart or reboot.

**ui.menu.item.background**
- Set the color of the background for menu items.
- Null (default)
- #RRGGBB color codes
- Change causes system to restart or reboot.

**ui.menu.item.text.color**
- Set the color of the text for menu items.
- Null (default)
- #RRGGBB color codes
- Change causes system to restart or reboot.

**ui.menu.title.background**
- Set the background color for the title of the Menu.
- Null (default)
- #RRGGBB color codes
- Change causes system to restart or reboot.

**ui.softkey.background**
- Set the background color for softkeys.
- Null (default)
- #RRGGBB color codes
ui.softkey.text.color
Set the color of the text for softkeys.
Null (default)
#RRGGBB color codes
Change causes system to restart or reboot.

ui.statusBar.background
Set the background color for the Status bar.
Null (default)
#RRGGBB color codes
Change causes system to restart or reboot.

ui.statusBar.text.color
Set the color of the text that displays on the Status bar.
Null (default)
#RRGGBB color codes
Change causes system to restart or reboot.

Default Phone Screen
Configure the default phone screen that displays when the phone is off-hook or in an active call.

Off-Hook Phone Screen
When the phone goes off-hook, you can configure the phone to display either the dialer view or the Lines screen by default.

If the Lines screen is set as the default; when the user dials a number, the dialer screen displays. If the user selects the New Call soft key, the line screen displays.

Displaying the Lines screen when the phone goes off-hook enables users to quickly select a favorite or BLF line to dial. In this scenario, when users start to enter a number from the keypad, the phone switches to the dialer screen.

Off-Hook Phone Screen Parameter

up.OffHookLineView.enabled
Set the default screen that appears when the phone goes off-hook.
0 (Default) - Dialing screen
1 - Line screen
up.offHookSpeedDialShortcut.enable

1 (default) - Displays the speed dial shortcut for one or two digits followed by # in the off-hook state.
0 – Does not display the speed dial shortcut for one or two digits followed by # in the off-hook state.

Active Call Phone Screen

In an active call, you can configure the phone to display the active call screen or the Lines screen.

You can configure the phones to display the screens as follows:

- The normal active call screen or call overlay showing active call information.
- The Lines screen, showing active call information in the ribbon at the top of the screen.

Displaying the Lines screen during an active call enables users to see the status of any lines, buddies, and BLF contacts they are monitoring without active call information getting in the way.

It is still possible to switch between the normal active call display and the lines view regardless of the default screen you set.

Active Call Screen Parameters

Use the parameters in the following list to set the default screen that displays when the phone is in call.

up.LineViewCallStatus.enabled

0 (default) - In an active call, the active call screen displays. Any incoming or outgoing call triggers the display of the active call screen.
1 - During an incoming call and in an active call, the line view displays and call details display on the status ribbon.

up.LineViewCallStatus.timeout

Specify the timeout period in seconds after which the phones go back to the Line Screen when the user goes to the Active Call Screen from the Line View.

10 (default)
2 - 10

Graphic Display Background

You can replace the phone’s default background image with a custom image or import multiple images that users can select from.

Poly phones support JPEG, BMP, and PNG image file formats. The phone doesn’t support progressive/ multi-scan JPEG images.
Maximum Image and Logo Sizes

Refer to the following table for the maximum image size supported for each phone model.

<table>
<thead>
<tr>
<th>Phone</th>
<th>Screen Size</th>
<th>Recommended Logo Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCX 400</td>
<td>720 x 1280</td>
<td>135 x 135</td>
</tr>
<tr>
<td>CCX 500</td>
<td>720 x 1280</td>
<td>135 x 135</td>
</tr>
<tr>
<td>CCX 600</td>
<td>1024 x 600</td>
<td>135 x 135</td>
</tr>
<tr>
<td>CCX 700</td>
<td>1024 x 600</td>
<td>135 x 135</td>
</tr>
</tbody>
</table>

Graphic Display Background Parameters

The configured background image displays across the entire phone screen, and the time, date, line and key labels display over the background.

If you want the background image to display more visibly from behind line key labels, use `up.transparentLines` to render line key labels transparent.

Use the parameters in the following list to configure graphic display background.

- **bg.background.enabled**
  
  Enable or disable the ability for users to set a custom background image on the phone screen. If enabled, options for customization are available on the phone screen and in the Web Configuration Utility for users.
  
  0 (default) - Disabled
  
  1 - Enabled

- **bg.color.bm.x.name**
  
  Specify the name of the phone screen background image file including extension with a URL or file path of a BMP or JPEG image.

  **Note:** If the file is missing or unavailable, the built-in default solid pattern is displayed.

- **up.transparentLines**
  
  0 (Default) - Line keys block display of the background image.
  
  1 - Line keys are transparent and allow the background image to display behind the line labels.
Digital Picture Frame

Connect a USB with images and display a slide show on the phone's idle screen.

For images to display, save the images in JPEG, BMP, or PNG format. Place the image in the root directory of the USB storage device. The phone can display a maximum image size of 9999x9999 pixels and a maximum of 1000 images.

The phone supports 9999x9999 images and progressive/multi-scan JPEG images. The maximum image size depends on the available memory in the phone.

You can access the digital picture frame on the web using PicFrame:// URL.

Digital Picture Frame Parameters

The parameters you can configure are included in the following list.

**feature.pictureFrame.enabled**

Enable or disable the digital picture frame.

- 1 (default) - Enabled
- 0 - Disabled

Change causes system to restart or reboot.

**up.pictureFrame.folder**

Path name for images.

- NULL (Default) - Images stored in the root folder on the USB flash drive are displayed.
- string - 0 to 40 characters

Example: If images are stored in the /images/phone folder on the USB flash drive, set this parameter to *images/phone*.

**up.pictureFrame.timePerImage**

Specify the number of seconds to display each picture frame image before moving to the next picture.

- 5 (Default)
- 3-300

Background Image Lock

By default, users can set a background image for their phones using the phone, a USB drive attached to the phone, or the Web Configuration Utility.

You can disable the user's ability to set images as a background when viewing images on a USB attached to the phone.

Disabling this feature removes the following options for users:
Phone Languages

All phones support the following languages: Arabic, Simplified Chinese, Traditional Chinese, Danish, Dutch, English, French, German, Italian, Japanese, Korean, Norwegian, Polish, Brazilian Portuguese, Russian, Slovenian, International Spanish, and Swedish.

Each language is stored as a language file in the VVXLocalization folder, which is included with the UC Software package. If you want to edit the language files, you must use a Unicode-compatible XML editor such as XML Notepad 2007 and familiarize yourself with the guidelines on basic and extended character support.

At this time, the updater is available in English only.

Phone Language Parameters

You can select the language that displays on the phone using the parameters in the following list.

device.spProfile

Set the default language that displays on the phone.

- NULL (default) - The default language is an empty string (lcl.ml.lang=""), which is English.
- DT - The default language is German (lcl.ml.lang="DTGerman_Germany").

lcl.ml.lang

- Null (default) - Sets the phone language to US English.
- String - Sets the phone language specified in the lcl.ml.lang.menu.x.label parameter.

lcl.ml.lang.menu.x

- Specifies the dictionary files for the supported languages on the phone. Dictionary files must be sequential. The dictionary file cannot have capital letters, and the strings must exactly match a folder name of a dictionary file.
- Null (default)
- String

lcl.ml.lang.menu.x.label

- Specifies the phone language menu label. The labels must be sequential.
- Null (default)
- String
**Multilingual Parameters**

The multilingual parameters included in the following list are based on string dictionary files downloaded from the provisioning server. These files are encoded in XML format and include space for user-defined languages.

- **lcl.ml.lang.clock.x.24HourClock**
  1 (default) - Displays the time in 24-hour clock mode.
  0 - Does not display the time in 24-hour clock mode.
  
  **Note:** Overrides the `lcl.datetime.time.24HourClock` parameter.

- **lcl.ml.lang.clock.x.dateTop**
  1 (default) - Displays date above time.
  0 - Displays date below time.
  
  **Note:** Overrides the `lcl.datetime.date.dateTop` parameter.

- **lcl.ml.lang.clock.x.format**
  "D,dM" (default)
  String
  The field may contain 0, 1 or 2 commas which can occur only between characters and only one at a time.
  For example: D,dM = Thursday, 3 July or Md,D = July 3, Thursday.
  
  **Note:** Overrides the `lcl.datetime.date.format` parameter to display the day and date.

- **lcl.ml.lang.clock.x.longFormat**
  1 (default) - Displays the day and month in long format (Friday/November).
  0 - Displays the day and month in abbreviated format (Fri/Nov).
  
  **Note:** Overrides the `lcl.datetime.date.longFormat` parameter.

- **lcl.ml.lang.japanese.font.enabled**
  Enable or disable the use of Japanese kana format.
  0 (default) - Disabled
  1 - Enabled
  Change causes system to restart or reboot.
**lcl.ml.lang.list**

Displays the list of languages supported on the phone.

- All (default)
- String

Change causes system to restart or reboot.

The basic character support includes the Unicode character ranges listed in the next table.

### Unicode Ranges for Basic Character Support

<table>
<thead>
<tr>
<th>Name</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>C0 Controls and Basic Latin</td>
<td>U+0000 - U+007F</td>
</tr>
<tr>
<td>C1 Controls and Latin-1 Supplement</td>
<td>U+0080 - U+00FF</td>
</tr>
<tr>
<td>Cyrillic (partial)</td>
<td>U+0400 - U+045F</td>
</tr>
</tbody>
</table>

**Add a Language for the Phone Display and Menu**

Use the multilingual parameters to add a new language to your provisioning server directory to display on the phone screen and menu.

### Procedure

1. Create a new dictionary file based on an existing one.
2. Change the strings making sure to encode the XML file in UTF-8 but also ensuring the UTF-8 characters chosen are within the Unicode character ranges indicated in the tables below.
3. Place the file in an appropriately named folder according to the format `language_region` parallel to the other dictionary files under the VVXLocalization folder on the provisioning server.
4. Add an `lcl.ml.lang.clock.menu.x` parameter to the configuration file.
5. Add `lcl.ml.lang.clock.x.24HourClock`, `lcl.ml.lang.clock.x.format`, `lcl.ml.lang.clock.x.longFormat`, and `lcl.ml.lang.clock.x.dateTop` parameters and set them according to the regional preferences.
6. (Optional) Set `lcl.ml.lang` to be the new `language_region` string.

**Pinyin Text Input**

Pinyin is the phonetic system used to transcribe Mandarin pronunciation of Chinese into Latin characters. Pinyin uses [Nuance XT9 Smart Input](#) to enable Chinese character input into phone text fields.

To enable users to use the pinyin, download a license key to the phone.
Hide the MAC Address
You can configure the phone to hide MAC address on the phone's display. When you enable this feature, users cannot view or retrieve the MAC address from the phone. The MAC address is available to administrators only.

Hide MAC Address Parameters
The following list includes parameters that configure the display of MAC address.

device.mac.hide.set
   Enable or disable the device.mac.hide parameter to control the display of MAC address information of phones to users.
   Null (default)
   0 - Disabled
   1 - Enabled

device.mac.hide
   0 (default) - MAC address displays.
   1 - MAC address is hidden.

Digital Phone Label
Configure the Digital Phone Label feature to display the complete registration line address in the status bar.

Digital Phone Label Parameters
You can create a short personal message to display in the status bar on the phone's screen.

lcl.status.LineInfoAtTop
   Enable or disable the text set in lcl.status.LineInfoAtTopText to display on the phone screen
   0 (default) - Disabled
   1 - Enabled

lcl.status.LineInfoAtTopText
   Provides the text be displayed on the phones screen. Up to 14 digits is allowed. The use of characters is permitted but might lead to truncation.
   Null (default)
   string
Note: You must enable lcl.status.LineInfoAtTop to configure this parameter.

Unique Line Labels for Registration Lines

You can configure unique labels on line keys for registration lines. You must configure multiple line keys on the phone for a registration in order to configure unique line labels. For example, you can set different names to display for the registration 4144 that displays on four line keys.

If you configure the line to display on multiple line keys without a unique label assigned to each line, the lines are labeled automatically in numeric order. For example, if you have four line keys for line 4144 labeled Poly, the line keys are labeled as 1_Poly, 2_Poly, 3_Poly, and 4_Poly. This also applies to lines without labels.

Unique Line Labels for Registration Lines Parameters

When using this feature with the parameter reg.x.label.y where x=2 or higher, multiple line keys display for the registered line address.

reg.x.line.y.label

Configure a unique line label for a shared line that has multiple line key appearances. This parameter takes effect when up.cfgUniqueLineLabel=1. If reg.x.linekeys=1, this parameter does not have any effect.

x = the registration index number starting from 1.

y = the line index from 1 to the value set by reg.x.linekeys. Specifying a string sets the label used for the line key registration on phones with multiple line keys.

If no parameter value is set for reg.x.line.y.label, the phone automatically numbers multiple lines by prepending "<y>" where <y> is the line index from 1 to the value set by reg.x.linekeys.

up.cfgLabelElide

Controls the alignment of the line label. By default when the line label is an alphanumeric or alphabetic string, the label aligns right. When the line label is a numeric string, the label aligns left.

None (Default)

Right

Left

up.cfgUniqueLineLabel

Allow unique labels for a registration that is split across multiple line keys using reg.X.linekeys.

0 (Default) - Use the same label on all line keys.

1 - Display a unique label as defined by reg.X.line.Y.label.
If `reg.X.line.Y.label` is not configured, then a label of the form `<integer>_` will be applied in front of the applied label automatically.

**LED Indicators**

LED indicators alert users to the different states of the phone and remote contacts. You can turn LED indicators on or off, set the pattern, color, and duration of a pattern for all physical keys on the phones.

You can set the pattern, color, and duration for the following LED indicators:
- Line keys
- Message Waiting Indicator (MWI)
- Headset key

**LED Behavior Parameters**

The LED pattern parameters listed in the following list configure the pattern state, color, and duration of the LED indicators and the pattern types on Poly devices.

For each parameter, specify `x`, `y`, and a permitted value:
- Specify an LED pattern using the LED pattern parameters.
- For `x`, specify an LED pattern type.
- For `y`, specify the step in the LED pattern with a number between 1-20.

Use the parameters in the following list to set the pattern state, color, and duration of the LED indicators.

**ind.pattern.x.step.y.state**
- 0 (default) - Turn off the LED indicator.
- 1 - Turn on the LED indicator.

**ind.pattern.x.step.y.color**
- Specify the color of the LED indicator.
  - Red (default)
  - Green
  - Yellow

**ind.pattern.x.step.y.duration**
- Specify the duration of the pattern in milliseconds.
  - 0 (default)
  - 0 - 32767
LED Indicator Pattern Types
Enter one of the values in the following table to indicate the LED indicator pattern type.

### LED Indicator Pattern Type

<table>
<thead>
<tr>
<th>Pattern Type</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>powerSaving</td>
<td>Sets the behavior for Message Waiting Indicator when the phone is in Power Saving mode.</td>
</tr>
<tr>
<td>active</td>
<td>Sets the pattern for line keys during active calls.</td>
</tr>
<tr>
<td>on</td>
<td>Turns on the LED indicator pattern.</td>
</tr>
<tr>
<td>off</td>
<td>Turns off the LED indicator pattern.</td>
</tr>
<tr>
<td>offering</td>
<td>Sets the pattern for line keys during incoming calls.</td>
</tr>
<tr>
<td>flash</td>
<td>Sets the pattern for line keys during held calls and the Message Waiting Indicator when there are unread voicemail messages.</td>
</tr>
<tr>
<td>lockedOut</td>
<td>Sets the pattern for line keys when a remote party is busy on a shared line.</td>
</tr>
<tr>
<td>held</td>
<td>Sets the pattern for line keys during a held call.</td>
</tr>
<tr>
<td>remoteBusyOffering</td>
<td>Sets the pattern for line keys for monitored BLF contacts when the BLF is in an active call and receives a new incoming call.</td>
</tr>
<tr>
<td>blfHold</td>
<td>Sets the pattern for BLF line keys when a call is on the hold. The default pattern is slow flashing red color LED.</td>
</tr>
<tr>
<td>parkedCallSelf</td>
<td>Sets the LED pattern for a self-parked call.</td>
</tr>
<tr>
<td>parkedCallRemote</td>
<td>Sets the LED pattern for remote-parked call.</td>
</tr>
</tbody>
</table>

LED Pattern Examples
This section includes example configurations you can use to set the patterns of LED indicators.

**Example: Disable the Headset Key LED in Headset Memory Mode**
By default, the Headset key glows green for analog headsets and blue for USB headsets.
The Headset key also flashes by default if Headset Memory Mode is enabled.
You can disable and turn off the flash pattern for the Headset key when Headset Memory Mode is enabled.

**Procedure**
» Set the parameter `ind.pattern.flashSlow.step.1.state` to 0.

**Example: Set an LED Pattern for Active Calls**
In the following example, during an active call, the line key alternates green and red.
Procedure

- Configure the pattern as follows:
  - `ind.pattern.active.step.1.color= "Green"
  - `ind.pattern.active.step.1.state= "1"
  - `ind.pattern.active.step.1.duration= "1000"
  - `ind.pattern.active.step.2.color= "Red"
  - `ind.pattern.active.step.2.state= "1"
  - `ind.pattern.active.step.2.duration= "1000"

Example: Set an LED Pattern for BLF Hold calls

In the following example, when monitored BLF line is on hold, the LED indicator changes to slow flashing red.

By default, the following parameters set the behavior of the BLF hold line key LED indicators.

`ind.pattern.blfHold.step.1.state`

- 0 – Turns off the LED indicator for BLF Hold.
- 1 (default) – Turns on the LED indicator for BLF Hold.
  
  Change causes system to restart or reboot.

`ind.pattern.blfHold.step.1.duration`

Specify the duration of the LED indicator for the pattern when BLF is in a hold state.

- 1000 (default)
- 0- 32767
  
  Change causes system to restart or reboot.

`ind.pattern.blfHold.step.1.color`

Set the color of the LED indicator for the pattern when BLF is in a hold state.

- Red (default) – LED indicator turns to red when the BLF is in a hold state.
- Green – LED indicator turns green when the BLF is in hold state.
  
  Change causes system to restart or reboot.

`ind.pattern.blfHold.step.2.state`

- 0 (default) – Turns off the LED indicator for BLF Hold.
- 1– Turns on the LED indicator for BLF Hold.
  
  Change causes system to restart or reboot.

`ind.pattern.blfHold.step.2.duration`

Specify the duration of the LED indicator for the pattern when BLF is in a hold state.

- 1000 (default)
0 - 32767
Change causes system to restart or reboot.

`ind.pattern.blfHold.step.2.color`
Set the color of the LED indicator for the pattern when BLF is in a hold state.
- Red (default) – LED indicator turns to red when the BLF is in a hold state.
- Green – LED indicator turns green when the BLF is in hold state.
Change causes system to restart or reboot.

**Example: Turn Off the Message Waiting Indicator in Power Saving Mode**
When Power Saving mode is enabled, the screen darkens, and the MWI flashes red.
By default, the powerSaving pattern has two steps before the pattern is repeated: a quick on period and then a long off period.
You can turn off the MWI or change the duration of the pattern steps.

**Procedure**

» Set the parameter `ind.pattern.powerSaving.step.1.state` to 0.

**Example: Change the Color of Line Key Indicators for Incoming Calls**
When a phone receives an incoming call, the line key LED indicator flashes green.
You can change the color of the indicator to Yellow or Red for incoming calls.

**Procedure**

» Set the parameter `ind.pattern.offering.step.1.color` to Yellow.

**LED Patterns For Self-Parked Calls**
The LED pattern for self-parked calls is solid red. Use the following parameters to configure the LED pattern for self-parked calls.

`ind.pattern.parkedCallSelf.step.x.color`
- Sets the LED color indicator for self-parked call.
  - Solid red (default)

`ind.pattern.parkedCallSelf.step.x.state`
- Controls the LED indicator for self-parked call.
  - 1 (default) - Turns on the LED indicator.
  - 0 – Turns off the LED indicator.

`ind.pattern.parkedCallSelf.step.x.duration`
- Specifies the duration of the LED indicator for self-parked call.
  - 500 (default)
LED Patterns for Remote-Parked Calls

The LED pattern for remote-parked calls is blinking red. Use the following parameters to configure the LED pattern for remote-parked calls.

- **ind.pattern.parkedCallRemote.step.x.color**
  
  Sets the LED color for remote-parked call.
  
  Blinking red (default)

- **ind.pattern.parkedCallRemote.step.x.state**
  
  Controls the LED indicator for the remote-parked call.
  
  1 (default) - Turns on the LED indicator.
  0 – Turns off the LED indicator.

- **ind.pattern.parkedCallRemote.step.x.duration**
  
  Specifies the duration of the LED indicator for the remote-parked call.
  
  500 (default)

Capture Your Phone’s Screen

You can capture your phone’s current screen.

Before you can take a screen capture, make sure the phone’s web server is enabled.

Procedure

1. Add the parameter `up.screenCapture.enabled` to your configuration.
2. Set the value to 1 and save.
3. On the device, go to Settings > Basic > Preferences > Screen Capture.
   
   Note you must repeat this step each time the device restarts or reboots.
4. Locate and record the phone’s IP address at Status > Platform > Phone > IP Address.
5. Set the phone to the screen you want to capture.
6. In a web browser address field, enter `https://<phoneIPaddress>/captureScreen` where `<phoneIPaddress>` is the IP address you obtained from the phone.
7. Enter the username Polycom and the phone’s current password.

   The web browser displays an image showing the phone’s current screen. You can save the image as a .bmp or .jpeg file.

Capture Your Device’s Current Screen Parameters

Use the following parameters to get a screen capture of the current screen on your device.

- **up.screenCapture.enabled**
  
  0 (Default) - The Screen Capture menu is hidden on the phone.
The Screen Capture menu displays on the phone. When the phone reboots, screen captures are disabled from the Screen Capture menu on the phone. Change causes system to restart or reboot.

**up.screenCapture.value**

0 (Default) - The Screen Capture feature is disabled.
1 - The Screen Capture feature is enabled.

**Line View Pages**

Polycom UC software supports line views in multiple pages.

You can navigate to a maximum of four pages when using this feature. When using page navigation, the page indicator and the corresponding LED line key highlight for an incoming call, and the phone performs the following actions:

- If the phone receives multiple calls or notifications, the page indicator highlights the pages in a sequential order from the first page. The precedence order for the page indicator is 1, 2, 3, and 4.
- If you are on the page with an active call, the LED for the line blinks. If you navigate to a different page during the call, the LED used for the same line key doesn’t blink.

Keep in mind the following phone behaviors:

- You can’t navigate between pages when you enable pagination and try to transfer or forward an active call.
- In idle browser view, the page indicator highlights the active page when you press the **Next** softkey.
- Press the **Home** key or swipe right on the screen to switch from idle browser view to line view.
- Pagination disables when you enable the Show only registration line.

**Navigate Line Screen Pages**

You can navigate through pages on the line screen.

**Procedure**

» Do one of the following:
  • Swipe the screen left or right.

**Pagination Configuration Parameter**

Use the following parameter to configure pagination on the phone.

**up.Pagination.enabled**

Enable or disable the pagination feature.

0 (default) - Disable
1 - Enable
This section explains features you can use to customize phones.

**Microbrowser and Web Browser**

The microbrowser and web browser include a Server Name Indication (SNI) add-on. SNI allows secure websites to present multiple certificates on the same IP address and TCP port.

**Note:** The exact functions and performance of the microbrowser and web browser vary by phone model.

For more information on creating applications for the phones, see the *Polycom Web Application Developer's Guide* at Polycom UC Software Support Center.

**Note:** The browser restarts in the following situations:

- The browser uses over 30MB of memory.
- The amount of free memory on the phone is below 6MB.
- The time is between 1am to 5am.

After the browser restarts, the last displayed web page restores.

**Microbrowser and Web Browser Parameters**

You can configure the microbrowser and web browser to display a non-interactive web page on the phone's idle screen, and you can specify an interactive home web page that users can launch in a web browser.

The parameters listed below configure the home page, proxy, and size limits used by the microbrowser and browser when selected to provide services.

**apps.push**

Specify the push server settings, including message type, port, tunnel, and a user name and password.
**apps.push.alertSound**

Enable for the phone to chime a sound when an alert is pushed.

0 (default) - Disabled
1 - Enabled

**apps.push.messageType**

Choose a priority level for push messages from the application server to the phone.

0 (None) - (default) - Discard push messages
1 (Normal) Allows only normal push messages
2 (Important) Allows only important push messages
3 (High) Allows only priority push messages
4 (Critical) Allows only critical push
5 (All) Allows all push messages

**apps.push.password**

The password to access the push server URL.

NULL (default)

string

**apps.push.secureTunnelEnabled**

Enable to allow the connection to the web server to use a secure tunnel.

1 (default) - Enabled
0 - Disabled

**apps.push.secureTunnelPort**

Specify the port the phone uses to communicate to the web server when the secure tunnel is used.

443 (default)
1 - 65535

**apps.push.secureTunnelRequired**

Enable for communications to the web server require to require a secure tunnel.

1 (default) - Enabled
0 - Disabled

**apps.push.serverRootURL**

The URL of the application server you enter here is combined with the phone address and sent to the phone's browser. For example, if the application server root URL is `http://172.24.128.85:8080/sampleapps`
and the relative URL is /examples/sample.html, the URL sent to the microbrowser is http://172.24.128.85:8080/sampleapps/examples/sample.html. You can use HTTP or HTTPS.

NULL (default)

URL

**apps.push.username**

The user name to access the push server URL. To enable the push functionality, you must set values for the parameters **apps.push.username** and **apps.push.password** (not null).

NULL (default)

string

**apps.statePolling**

Specify phone state polling settings, such as response mode, the poll URL, and a user name and password.

**apps.statePolling.password**

Enter the password that the phone requires to authenticate phone state polling.

NULL (default)

string

**apps.statePolling.responseMode**

1 (default) - Polled data you request is sent to a configured URL.

0 - Polled data is sent in the HTTP response.

**apps.statePolling.URL**

The URL to which the phone sends call processing state/device/network information. The protocol used can be either HTTP or HTTPS. Note: To enable state polling, the parameters **apps.statePolling.URL**, **apps.statePolling.username**, and **apps.statePolling.password** must be set to non-null values.

NULL (default)

string

**apps.statePolling.username**

Enter the user name that the phone requires to authenticate phone state polling.

NULL (default)

string

**apps.telNotification.appInitializationEvent**

0 (default) - No telephony notification event is sent.
1 - An XML telephony notification event is sent to report that the phone has completed initialization of its primary UC Software application. This event typically means that the phone is available and ready to receive network requests even if the phone user interface is not yet available.

**apps.telNotification.callStateChangeEvent**

0 (default) - Call state change notification is disabled.
1 - Call state notification is enabled.

**apps.telNotification.incomingEvent**

0 (default) - Incoming call notification is disabled.
1 - Incoming call notification is enabled.

**apps.telNotification.lineRegistrationEvent**

0 (default) - Line registration notification is disabled.
1 - Line registration notification is enabled.

**apps.telNotification.networkUpEvent**

0 (default) - No telephony notification event is sent.
1 – An XML telephony notification event is sent to report that the phone has received link up state from its LAN port and that an IP address was assigned.

**apps.telNotification.offhookEvent**

0 (default) - Disable off-hook notification.
1 - Enable off-hook notification.

**apps.telNotification.onhookEvent**

0 (default) - Disable on-hook notification.
1 - Enable on-hook notification.

**apps.telNotification.outgoingEvent**

0 (default) - Disable outgoing call notification.
1 - Enable outgoing call notification.

**apps.telNotification.taInitializationEvent**

0 (default) – No telephony notification event is sent.
1 - An XML telephony notification event is sent to report that the phone has started its test automation server and is ready to receive API commands.

**apps.telNotification.uiInitializationEvent**

0 (default) - No telephony notification event is sent.
1 - An XML telephony notification event is sent to report that the phone has completed start up of the phone user interface and is ready to receive physical key or touch inputs.

**apps.telNotification.URL**

The URL to which the phone sends notifications of specified events. You can use HTTP or HTTPS.

NULL (default)

string

**apps.telNotification.userLogInOutEvent**

Enable or disable the user login/logout notification.

0 (default) - Disabled
1 - Enabled

**apps.telNotification.x.URL**

The URL to which the phone sends notifications of specified events, where x 1 to 9. You can use HTTP or HTTPS.

NULL (default)

string

**mb.main.home**

Specifies the URL of the microbrowser's home page. For example: http://www.example.com/xhtml/frontpage/home.

Null (default)

valid HTTP URL, String (maximum 255 characters)

**mb.main.idleTimeout**

Specifies the timeout in seconds for the interactive browser. If the interactive browser remains idle for a defined period of time, the phone returns to the idle browser. If set to 0, there is no timeout.

40 (default)

0 - 600

**mb.main.loadWebImages**

Enable to allow images to load in the web browser.

1 (default) - Enabled
0 - Disabled

**mb.main.proxy**

Specifies the address of the HTTP proxy to be used by the microbrowser.

Null (port: 8080) (default)
domain name or IP address in the format <address>:<port>

mb.main.reloadPage
0 (default) - The microbrowser displays the content of the most recently viewed web page
1 - The microbrowser loads the URL configured in mb.main.home each time the browser is launched.

mb.main.statusbar
0 (default) - The status bar does not get displayed.
1 - The status bar and status messages are displayed.

mb.main.toolbar.autoHide.enabled
1 (default) - The toolbar is not displayed.
0 - The toolbar displays continuously.

mb.proxy
Specify the Application browser home page, a proxy to use, and size limits.

Support for REST API
Poly phones support REST APIs that enable you to execute certain functions and retrieve information.
For more information on phone APIs, see REST API Reference Manual for UC Software at the Polycom Support Site.
The REST API feature is disabled by default. You can use parameters to enable REST API on your phone.

REST API Parameter
Use the following parameter to enable the REST API.

apps.restapi.enabled
0 (default) - Disabled
1 - Enabled

Soft Keys
You can create custom soft keys that enable users to access frequently used functions, create menu shortcuts to frequently used phone settings, or create a soft key in place of a hard key not available on the phone.
For example, if the phone does not have a Do Not Disturb hard key, you can create a Do Not Disturb soft key.
You can create custom soft keys as any of the following:

- An enhanced feature key sequence
- A speed dial contact directory entry
- An enhanced feature key macro
- A URL
- A chained list of actions

**Call State for Custom Soft Keys**

You can configure soft keys to display certain functions depending on the phone's menu level or call state.

For example, you can make a Call Park soft key available when the phone is in an active call state.

You can configure custom soft keys to display for the following call states:

- **Idle** – There are no active calls.
- **Active** – This state starts when a call is connected. It stops when the call stops or changes to another state (like hold or dial tone).
- **Alerting (or ringing or incoming proceeding)** – The phone is ringing.
- **Dial tone** – You can hear a dial tone.
- **Proceeding (or outgoing proceeding)** – This state starts when the phone sends a request to the network. It stops when the call is connected.
- **Setup** – This state starts when the user starts keying in a phone number. This state ends when the Proceeding state starts.
- **Hold** – The call is put on hold locally.

**Soft Key Parameters**

You can create up to 10 custom soft keys.

If you configure more soft keys than what can fit on the phone's screen, a More soft key displays. Users can use the More soft key to display any additional soft keys available.

If you want the phone to display both default and custom soft keys, you can configure them in any order. However, the order in which soft keys display depends on the phone's menu level and call state. If you have configured custom soft keys to display with the default soft keys, the order of the soft keys may change.

**Note:** The Hold, Transfer, and Conference soft keys are grouped together to avoid usability issues. You may experience errors if you try to insert a soft key between these three grouped soft keys.

The following list includes the parameters for configuring soft keys. Note that this feature is part of enhanced feature keys (EFK), and you must enable the EFK parameters to configure soft keys. See the Enhanced Feature Keys section for details about configuring soft keys and line keys.

**feature.enhancedFeatureKeys.enabled**

- 0 (default) - Disables the enhanced feature keys feature.
- 1 - Enables the enhanced feature keys feature.
**softkey.x.action**

Controls the action or function for the custom soft key x.

Null (default)

macro action string, 2048 characters

This value uses the same macro action string syntax as an Enhanced Feature Key.

**softkey.x.enable**

0 (default) - The x soft key is disabled.

1 - The x soft key is enabled.

**softkey.x.insert**

0 (default) - The phone places the soft key in the first available position.

0 to 10 - The phone places the soft key in the corresponding position and moves the following soft keys by one position to the right.

For example, if the soft key is set to 3, the soft key is displayed in the third position from the left. If the soft key already exists in the third position, it is moved to fourth position and the following soft keys are moved to right by one space.

If `softkey.x.precede` is configured, this value is ignored. If the insert location is greater than the number of soft keys, the key is positioned last after the other soft keys.

**softkey.x.label**

The text displayed on the soft key label. If Null, the label is determined as follows:

- If the soft key performs an Enhanced Feature Key macro action, the label of the macro defined using `efk.efklist` is used.
- If the soft key calls a speed dial, the label of the speed dial contact is used.
- If the soft key performs chained actions, the label of the first action is used.
- If the soft key label is Null and none of the preceding criteria are matched, the label is blank.

Null (default)

String

**Note:** The maximum number of characters for this parameter value is 15; however, the maximum number of characters that a phone can display on its user interface varies by phone model and by the width of the characters used. Parameter values that exceed the phone’s maximum display length are truncated by ellipses (...). The phone truncates the beginning of numerical labels (for example, …4567) and truncates the end of alphabetical labels (for example, Abcd…).

**softkey.x.precede**

0 (default) - The phone locates the soft key in the first available position from left.

1 - The phone locates the soft key before the default soft key position.
**softkey.x.use**
Specify which call states the soft key displays in.

**softkey.x.use.active**
- 0 (default) - Does not display the soft key x during an active call.
- 1 - Displays the soft key x during an active call.

**softkey.x.use.alerting**
- 0 (default) - Does not display the soft key x in an alerting state during an active call.
- 1 - Displays the soft key x in an alerting state during an active call.

**softkey.x.use.dialtone**
- 0 (default) - Does not display the soft key in the dial tone state during an active call.
- 1 - Displays the soft key x in the dial tone state during an active call.

**softkey.x.use.hold**
- 0 (default) - Does not display the soft key x in the hold state during an active call.
- 1 - Displays the soft key x in the hold state during an active call.

**softkey.x.use.idle**
- 0 (default) - Does not display the soft key x in the idle state during an active call.
- 1 - Displays the soft key x in the idle state during an active call.

**softkey.x.use.park**
- 0 (default) - Does not display the soft key x in the parked state during an active call.
- 1 - Displays the soft key x in the parked state during an active call.

**softkey.x.use.proceeding**
- 0 (default) - Does not display the soft key x in the proceeding state during an active call.
- 1 - Displays the soft key x in the proceeding state during an active call.

**softkey.x.use.setup**
- 0 (default) - Does not display the soft key x in the setup state during an active call.
- 1 - Displays the soft key x in the setup state during an active call.

**softkey.feature.intercom**
- 1 (default) - Enables the Intercom soft key.
- 0 - Disables the Intercom soft key.
softkey.feature.doNotDisturb
1 (default) - Enables the DND soft key on the phone.
0 - Disables the DND soft key on the phone.

softkey.feature.buddies
1 (default) - Displays the Buddies soft key.
0 - Does not display the Buddies soft key.

softkey.feature.callers
0 (default) - Displays the Callers soft key for all platforms.
1 - Does not display the Callers soft key for all platforms.

softkey.feature.directories
1 (default) - Displays the Directories (Dir) soft key.
0 - Does not display the Directories (Dir) soft key.
Change causes system to restart or reboot.

softkey.feature.doNotDisturb
1 (default) - Enables the DND soft key.
0 - Disables the DND soft key.

softkey.feature.endcall
1 (default) - Displays the End Call soft key.
0 - Does not display the End Call soft key.

softkey.feature.forward
1 (default) - Displays the Forward soft key.
0 - Does not display the Forward soft key.

softkey.feature.join
1 (default) - Displays the Join soft key.
0 - Does not display the Join soft key.

softkey.feature.mystatus
1 (default) - Displays the MyStatus soft key (if pres.idleSoftKeys is set to 1).
0 - Does not display the MyStatus soft key.

softkey.feature.newcall
1 (default) - Displays the New Call soft key is displayed.
0 - Does not display the New Call soft key.

**softkey.feature.redial**

0 (default) - Displays the Redial soft key.
1 - Does not display the Redial soft key.

The parameter `feature.enhancedFeatureKeys.enabled` must be set to 1 first to configure this feature, and the parameter `efk.softkey.alignleft` must be set to 1 to move enabled soft keys into the positions of disabled soft keys.

**softkey.feature.split**

1 (default) - Displays the Split soft key to split the conference call to individual calls.
0 - Does not display the Split soft key.

---

**Soft Key Customization Parameters**

You can use the soft key parameters to customize soft keys on the phone interface.

---

**Note:** The parameter `feature.enhancedFeatureKeys.enabled` must be enabled (set to 1) to use the Configurable Soft Key feature.

---

In the following list of soft key configuration parameters, x can equal from 1-10 soft keys.

**softkey.feature.basicCallManagement.redundant**

Enable to display the Hold and Transfer soft keys.

1 (default) - Enabled
0 - Disabled

**softkey.feature.buddies**

Enable to display the Buddies soft key.

1 (default) - Enabled
0 - Disabled

**softkey.feature.callers**

Enable to display the Callers soft key for all platforms

1 - Enabled
0 (default) - Disabled

**softkey.featuredirectories**

Enable to display the Directories (Dir) soft key.

1 (default) - Enabled
0 - Disabled
Change causes system to restart or reboot.

**softkey.feature.doNotDisturb**
Enable or disable the Do Not Disturb (DND) soft key.
1 (default) - Enabled
0 - Disabled

**softkey.feature.endcall**
Enable to display the End Call soft key.
1 (default) - Enabled
0 - Disabled

**softkey.feature.forward**
Enable to display the Forward soft key.
1 (default) - Enabled
0 - Disabled

**softkey.feature.join**
Enable to display the Join soft key.
1 (default) - Enabled
0 - Disabled

**softkey.feature.mystatus**
Enable to display the MyStatus soft key. The `pres.idleSoftKeys` parameter must be set to 1.
1 (default) - Enabled
0 - Disabled

**softkey.feature.newcall**
Enable to display the New Call soft key.
1 (default) - Enabled
0 - Disabled

**softkey.feature.redial**
Enable to display the Redial softkey. The parameter `feature.enhancedFeatureKeys.enabled` must be set to 1 first to configure this feature, and the parameter `efk.softkey.alignleft` must be set to 1 to move enabled soft keys into the positions of disabled soft keys.
1 - Enabled
softkey.feature.split
Enable to display the Split soft key. The Split soft key allows you to split conference calls into individual calls.
1 (default) - Enabled
0 - Disabled

Disabling Default Soft Keys
You can disable the display of any of the following default soft key to make room for custom soft keys:
- New Call
- End Call
- Split
- Join
- Forward
- Directories
- MyStatus and buddies
- Hold, transfer, and conference

Example: Transfer Call to Broadsoft Voicemail
Use the following example configuration to automatically transfer an active call to a BroadSoft voicemail.

In this example, *55 is the star code for BroadSoft voicemail, and 8545 is the extension of the voicemail line the call transfers to. The exact star code to transfer the active call to voicemail depends on your call server.

Enabling the parameter softkey.1.use.active causes the soft key to display when a call becomes active on the line. When you press the soft key—labeled VMail in this example—the call is placed on hold and automatically transferred to a BroadSoft voicemail.

Procedure
1. Update the configuration file as follows:
   - softkey.1.label="VMail"
   - softkey.1.action="$FTtransfer$$Cpause1$$FDialpadStar$$FDialpad5$ $FDialpad5$$FDialpad8$$FDialpad5$$FDialpad4$$FDialpad5$$FSsoftkey1$"
   - softkey.1.enable="1"
   - softkey.1.use.active="1"
2. Reboot the phone.
   When an incoming call connects and becomes active, the VMail soft key displays.
Example: Send-to-Voicemail Prompt

Use the following example to enable users to enter a voicemail extension to transfer an active call to BroadSoft voicemail.

In this example, *55 is the star code used for BroadSoft voicemail. The exact star code to transfer the active call to voicemail depends on your call server.

Enabling the parameter softkey.1.use.active causes the soft key to display when a call becomes active on the line. When a user presses the soft key, the call is placed on hold and a field prompts the user to enter the extension of a voicemail line to transfer the call to. The efk.prompt* parameters control the numeric prompt field users enter the extension into.

Note that this example works only on line 1 of the phone.

Procedure

1. Update the configuration file as follows:
   - softkey.1.label="VMail"
   - softkey.1.action="^*55$P1N10$$Tinvite$"
   - softkey.1.enable="1"
   - softkey.1.use.active="1"
   - efk.efkprompt.1.label="Voice Mail"
   - efk.efkprompt.1.status="1"
   - efk.efkprompt.1.type="numeric"

2. Reboot the phone.
   When an incoming call connects and becomes active, the VMail soft key displays.

3. Press the VMail soft key.
   A field displays prompting you to enter an extension.

4. Enter the extension you want to transfer the call to.

5. Press the Enter soft key.

Example: Speed Dial Soft Key with a Pause

Use the following example to configure a soft key to automatically dial a number with a pause in the dialing sequence.

In this example, use $CpauseX$ where X is the number of seconds to pause—7 in this example. Adding this pause function enables users to automatically dial into a conference ID that requires an entry code after the conference call is connected.

Procedure

» Update the configuration file as follows:
   - softkey.1.label="VMail"
   - softkey.1.action="$S1$$Tinvite$$Cwc$$Cpause7$$FDialpad8$$FDialpad5$$FDialpad4$$FDialpad5$"
   - softkey.1.enable="1"
   - softkey.1.use.idle="1"
   - feature.enhancedFeatureKeys.enabled="1"
The values for this example are explained as follows:

- \$S1\$— Speed dial line 1
- \$S1\$\$Tinvite\$—The phone sends an invite to \$S1\$
- \$Cwc\$ —The phone waits for the call to connect
- \$Cpause7\$ —The phone waits for 7 seconds before dialing the remaining numbers
- \$FDialpad8\$\$FDialpad5\$\$FDialpad4\$\$FDialpad5\$ —The phone enters the entry code 8545.

**Example: Directory-Linked Speed Dial Soft Key with a Pause**

Use the following example to add a speed dial line key linked to a directory file with a pause in the dialing sequence.

**Procedure**

1. Update the configuration file as follows:
   - \texttt{feature.enhancedFeatureKeys.enabled="1"}
   - \texttt{efk.efklist.1.action.string="501$Tinvite$$Cwc$$Cpause7$1234#$Tdtmf$"}
   - \texttt{efk.efklist.1.label="number"}
   - \texttt{efk.efklist.1.mname="number"}
   - \texttt{efk.efklist.1.status="1"}

2. In a contact directory file or speed dial file (000000000000-directory.xml or MACaddress-directory.xml), add the following:
   - \texttt{<fn>Call Number</fn>}
   - \texttt{<ct>!number</ct>}
   - \texttt{<sd>99</sd>}

The following values are included in the action string: \texttt{"501$Tinvite$$Cwc$$Cpause7$1234#$Tdtmf$"}:

- \texttt{501$Tinvite$} —Dial 501
- \texttt{$Cwc$} —Wait for the call to connect
- \texttt{$Cpause7$} —A seven second pause
- \texttt{1234#$Tdtmf$} —Send 1234 dual-tone multi-frequency

The following EFK commands are linked to the directory file:

- The parameter \texttt{efk.efklist.1.mname="number"} is linked to the speed dial contact \texttt{<ct>!number</ct>} of the directory file
- Use \texttt{<fn>Call Number</fn>} to define the name that displays on the key
- Use \texttt{<sd>99</sd>} to identify which directory entry to link to the key
Enhanced Feature Keys

Enhanced feature keys (EFK) enables you to customize the functions of a phone's line, soft, and hard keys to assign frequently used functions to keys or to create menu shortcuts to frequently used phone settings.

Enhanced feature key functionality is implemented using star code sequences like *89 and SIP messaging. Star code sequences that define EFK functions are written as macros that you apply to line and soft keys. The EFK macro language was designed to follow current configuration file standards and to be extensible (see Macro Definitions).

When this feature is enabled, and the user presses Lines soft key, all the lines on the home screen will appear. You can press any line key to initiate the call to that number.

In addition, you can configure an EFK as a line key allowing the users to execute the macro action defined to that line key. When this feature is enabled, all the EFK macros that are configured using efk.eklist parameter and has efk.eklist.x.status=1 will display as a line key. You can enable or disable this feature using configuration parameter or importing the configuration file using the Web Configuration Utility.

For example, configure the phone with the following configuration:

```plaintext
feature.enhancedFeatureKeys.enabled="1"
feature.EFKLineKey.enabled="1"
efk.eklist.1.mname="DND"
efk.eklist.1.status="1"
efk.eklist.1.action.string="$FDoNotDisturb$"
```

After you run and update configuration, the DND EFK will display as a line key. When you press the DND line key, Do Not Disturb functionality is executed.

In addition, you can use Flexible Line Keys feature for an EFK and assign to a line key that displays anywhere on the phone's screen. For more information, see Flexible Line Key Assignments.

Enhanced Feature Keys Parameters

The rules for configuring EFK for line keys, softkeys, and hard keys vary.

Note: You can include configuration file changes and enhanced feature key definitions in one configuration file. However, Poly recommends creating a new configuration file to make configuration changes.

Before configuring EFK, refer to Macro Definitions to become familiar with the macro language.
See the following list for the parameters you can configure and a brief explanation of how to use the contact directory to configure line keys.

`reg.x.callsPerLineKey`
- Set the maximum number of concurrent calls for a single registration x. This parameter applies to all line keys using registration x. If registration x is a shared line, an active call counts as a call appearance on all phones sharing that registration.
- This per-registration parameter overrides `call.callsPerLineKey`.
- 24 (default)
- 1 - 24

`feature.enhancedFeatureKeys.enabled`
- 0 (default) - Disables the enhanced feature keys feature.
- 1 - Enables the enhanced feature keys feature.

`feature.EFKLineKey.enabled`
- 0 (default) – Does not allow configuring EFK as a line key.
- 1 - Allows configuring EFK as a line key.
- Before you enable this parameter, set the parameter `feature.enhancedFeatureKeys.enabled` to 1.

`efk.efklist.x.action.string`
- The action string contains a macro definition of the action that the feature key performs.
- Null (default)
- String (maximum of 64 characters)
- If you enable EFK, this parameter must have a value (it cannot be Null).
- For a list of macro definitions and example macro strings, see Macro Definitions.
- Change causes system to restart or reboot.

`efk.efklist.x.label`
- The text string used as a label on any user text entry screens during EFK operation.
- Null (default) - Uses the Null string.
- String (maximum of 64 characters)
- If the label does not fit on the screen, the text shortens and appends with '…'.
- Change causes system to restart or reboot.

`efk.efklist.x.mname`
- The unique identifier used by the speed dial configuration to reference the enhanced feature key entry. Cannot start with a digit. Note that this parameter must have a value, it cannot be Null.
- expanded_macro (default)
String (maximum of 64 characters)
Change causes system to restart or reboot.

**efk.efklist.x.status**

- 0 (default) - Disables the key x.
- Null - Disables the key x.
- 1 - Enables the key x.

Change causes system to restart or reboot.

**efk.efklist.x.type**
Defines the SIP method.
- Invite (default) - Performs the required action using the SIP INVITE method.
- Null - default of INVITE is used.

This parameter is included for backwards compatibility. Do not use if possible. If **efk.x.action.string** contains types, this parameter is ignored.

Change causes system to restart or reboot.

**efk.efkprompt.x.label**
The prompt text on the user prompt screen.
- Null (default) - No prompt displays.
- String

If the label does not fit on the screen, the label shortens and ‘...’ appends.

Change causes system to restart or reboot.

**efk.efkprompt.x.status**
This parameter must have a value. It cannot be Null.
- 0 (default) - Disables the key.
- 1 - Enabled the key.

If a macro attempts to use a prompt that is disabled or invalid, the macro execution fails.

Change causes system to restart or reboot.

**efk.efkprompt.x.type**
The type of characters entered by the user.
- text (default) - The phone interprets characters as letters.
- numeric - The phone interprets characters as numbers.

If Null, **numeric** is used. If this parameter has an invalid value, this prompt, and all parameters depending on this prompt, are invalid.

**Note:** A mix of **numeric** and **text** is not supported.
 Change causes system to restart or reboot.

**efk.efkprompt.x.userfeedback**

The user input feedback method.
visible (default) - The text is visible.
masked - The text displays as asterisk characters (*). You can use this to mask password fields.

If this parameter has an invalid value it and all dependent parameters are invalid.

Change causes system to restart or reboot.

**efk.version**

The version of the EFK elements. This parameter is not required if there are no **efk.efklist** entries.

2 (default) - Supported version for SIP 3.1 and later.
1 - Supported version for or SIP 3.0.x or earlier.
Null - Disables the EFK feature.

Change causes system to restart or reboot.

**efk.softkey.alignleft**

Use this parameter to left-align softkeys and remove blank softkeys from the order.

0 (default)
1 - Left-aligns softkeys and removes blank softkeys from the order

**Note:** This parameter doesn’t work with custom softkeys.

Change causes system to restart or reboot.

**efk.efklist.x.lineLabel**

Specifies EFK line key label.
ALL (default)

Change causes system to restart or reboot.

### Some Guidelines for Configuring Enhanced Feature Keys

Use the following guidelines to help you to configure enhanced feature keys (EFKs) efficiently:

- Activation of EFK functions requires valid macro construction.
- All failures are logged in the phone's app logs at level 4 (Minor Error).
- If two macros have the same name, the first one is used and the subsequent one is ignored.
- A sequence of characters prefixed with "!" are parsed as a macro name. The exception is the speed dial reference, which starts with "!" and contains digits only.
- A sequence of characters prefixed with "^" is the action string.
- "'!" and "'^" macro prefixes cannot be mixed in the same macro line.
• The sequence of characters must be prefixed by either "!" or "^" to be processed as an enhanced feature key. All macro references and action strings added to the local directory contact field must be prefixed by either "!" or "^".

• Action strings used in soft key definitions do not need to be prefixed by "^". However, the "!" prefix must be used if macros or speed dials are referenced.

• A sequence of macro names in the same macro is supported (for example, "!m1!m2" ).

• A sequence of speed dial references is supported (for example, "112" ).

• A sequence of macro names and speed dial references is supported (for example, "!m112!m2" ).

• Macro names that appear in the local contact directory must follow the format "!<macro name>" , where <macro name> must match an <elklist> mname entry. The maximum macro length is 100 characters.

• A sequence of macros is supported, but cannot be mixed with other action types.

• Action strings that appear in the local contact directory must follow the format "^<action string>". Action strings can reference other macros or speed dial indexes. Protection against recursive macro calls exists (the enhanced feature keys fails after you reach 50 macro substitutions).

Contact Directory Macros

Because line keys and their functions are linked to fields in the contact directory file, you need to match the contact field (ct) in the directory file to the macro name field (mname) in the configuration file that contains the EFK parameters.

When you enter macro names to the contact field (ct) in the directory file, add the '!' prefix to the macro name. The template directory configuration file is named 000000000000-directory~.xml. To use this file, remove the tilde (~) from the file name.

Special Characters

Macro names and macro labels cannot contain these special characters. If they do, you may experience unpredictable behavior.

The following special characters are used to implement the enhanced feature key functionality:

• ! The characters following it are a macro name.

• ' or ASCII (0x27) This character delimits the commands within the macro.

• $ This character delimits the parts of the macro string. This character must exist in pairs, where the $ delimits the characters to be expanded.

• ^ This character indicates that the following characters represent the expanded macro (as in the action string).

• Macro names and macro labels cannot contain these special characters. If they do, you may experience unpredictable behavior.

Enhanced Feature Key Example Configurations

The following configurations shown in the below illustration were set in the features.

cfg file:

• For the efk.efklist.x.* parameters, the following configurations were applied:
  • Line key 1 has been assigned a Call Park address (1955) and line key 2 a call retrieve function.
  • The parameter acton.string shows the macro definition for these two functions.
- Status is enabled and a label has been specified to display next to the line key.
- The entry in the mname parameter corresponds to the contact (ct) field in the contact directory.

- For the efk.prompt.* parameters, the following configurations were applied:
  - Status is enabled.
  - The label on the user prompt has been defined as Enter Number: and this prompt displays on the phone screen.
  - The type parameter has been set to numeric to allow only numbers.
  - userfeedback is specified as visible, which enables users to see the numbers entered into the prompt.

Macro Definitions

The efk.efklist.x.action.string can be defined by macro actions, prompt macro substitution or an expanded macro.

Macro Actions

The action string is executed in the order it displays.

User input is collected before any action is taken. The action string can contain the fields shown in the following table.

<table>
<thead>
<tr>
<th>Action String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L&lt;label&gt;$</td>
<td>This is the label for the entire operation. The value can be any string including the null string (in this case, no label displays). This label is used if no other operation label collection method worked (up to the point where this field is introduced). Make this the first entry in the action string to be sure that this label is used; otherwise another label may be used and this one ignored.</td>
</tr>
<tr>
<td>digits</td>
<td>The digits to be sent. The appearance of this parameter depends on the action string.</td>
</tr>
</tbody>
</table>
### Action String

<table>
<thead>
<tr>
<th>Action String</th>
<th>Description</th>
</tr>
</thead>
</table>
| $C<command>$  | This is the command. It can appear anywhere in the action string. Supported commands (or shortcuts) include:  
  * hangup (hu)  
  * hold (h)  
  * waitconnect (wc)  
  * pause <number of seconds> (p <num sec>) where the maximum value is 10 |
| $Tconsult$    | An administrator uses this macro to execute a consultative transfer irrespective of the default transfer type. The input to this macro is given using star code sequences or prompt macro substitution. For example,  
  * Call Park - *68<Number>$Tconsult$  
  * Consultative Transfer using direct number - <Number>$Tconsult$  
  * Consultative Transfer using prompt - $P1N10$$Tconsult$ |
| $T<type>$     | The embedded action type. Multiple actions can be defined. Supported action types include:  
  * invite  
  * dtmf  
  * refer  
  * intercom  
  
  Poly recommends that you always define this field. If it isn’t defined, the supplied digits are dialed using INVITE (if no active call) or DTMF (if an active call). The use of refer method is call server dependent and may require the addition of star codes. |
| $M<macro>$    | The embedded macro. The <macro> string must begin with a letter. If the macro name isn’t defined, the execution of the action string fails. |
| $P<prompt num>N<num digits>$ | The user input prompt string. |
| $S<speed dial index>$ | The speed dial index. Only digits are valid. The action is found in the contact field of the local directory entry pointed to by the index. |
| $F<internal function>$ | An internal key function. |
| $A<internal function>$ | The internal key function. If you need to add a value to the macro use a parameter value before the macro definition.  
  * For example, 1$AVoiceMail$ - Voice mail is an internal function, which uses “1” as an input for the parameter. |
| URL           | A URL. Only one per action string is supported. |

### Prompt Macro Substitution

The macros provide a generic and easy way to manage and define the prompt to be displayed to the user, the maximum number of characters that the user can input, and the action that the phone performs after all user input has been collected.

Macros are case sensitive.
If a macro attempts to use a prompt that is disabled, the macro execution fails. A prompt is not required for every macro.

For example, the `efk.efklist.x.action.string` can be defined by a macro substitution string, `PnNn`, where the following applies:

- $Pn$ is the prompt $x$ as defined by `efk.efkprompt.x`.
- $Nn$ is the number of digits or letters that the user can enter. The value must be between 1 and 32 characters otherwise the macro execution fails. The user must press the **Enter** soft key to complete data entry.

**Expanded Macros**

Expanded macros are prefixed with the "^^" character and are inserted directly into the local directory contact (ct) field.

**Example Macros**

The action string `$Changup*$444*$P1N4*$Tinvite*$Cwaitconnect*$P2N3*$Cpause2*$Tdtmf$ $Changup$` is executed in order as follows:

1. The user is prompted for 4 digits. For example, 1234.
2. The user is prompted for 3 digits. For example, 567.
3. The user's active call is disconnected.
4. The string *444*1234 is sent using the INVITE method.
5. After connection, there is a two second pause, and then the string 567 is sent using DTMF dialing on the active call.
6. The active call is disconnected.

Because line keys and their functions are linked to fields in the directory file, the macro name you enter in `efk.list.x.mname` must match the name you enter to the contact (ct) field in the directory file. The macro name you enter in the (ct) field of the directory file must begin with the "!" prefix.

**Flexible Line Key Assignment**

You can enable users to assign a line key function to any line key on the phone.

By default, functions are assigned to line keys in succession—the order in which the line key displays on the phone. Flexible Line Keys (FLK) enables you to break that ordering and assign a line key function to a line key that displays anywhere on the phone’s screen. You can apply this feature to any line key function, including line appearance, speed dial, busy lamp field (BLF), presence, and Enhanced Feature Keys.

**Flexible Line Keys Parameters**

Line keys that you configure using this feature override the default line key assignments as well as any custom line key configurations you may have made.

To use this feature, you need to specify the function of each line key on the phone. You do this by assigning a `category` (`lineKey.x.category`) and an `index` (`lineKey.x.index`) to each line key, both of which are explained in the Enhanced Feature Key Example Configurations.

Use the parameters in the following list to configure this feature.
lineKey.reassignment.enabled
   Enable to specify at least two calls per line key.
   0 (default) - Disabled
   1 - Enabled

lineKey.x.category
   Specify the line key category.
   Unassigned (default)
   Line
   BLF
   EFK
   SpeedDial
   Presence

lineKey.x.index
   Specify the line key number (dependent on category).
   0 (default) - The index value for BLF or presence.
   0- 9999

up.staticBLF.FLKIndexRequired
   Enable to display the static BLF in the specified position.
   0 (default) – Disabled
   1 - Enabled

up.EFK.FLKIndexRequired
   Enable to display the EFK in the specified position.
   0 (default) – Disabled
   1 - Enabled

Assigning Busy Lamp Field (BLF) and Presence to Line Keys

Specific conditions apply when you assign BLF or presence to line keys.

If you are assigning BLF or presence to a line key, assign that line key to `index=0` to indicate automatic ordering. BLF and presence line keys are self-ordering, meaning that if you have these features assigned to multiple line keys, you can specify the location of the BLF or presence line key but not the order in which they display. For example, you can assign a BLF line key to index 1, 3, and 5 but you cannot specify how the contacts are ordered, which BLF contacts display on line keys 1, 3, and 5.

In addition, to assign BLF and presence to a line key, you need to assign a corresponding registration line. You can configure multiple line keys per registration if each line key has a corresponding `reg.x.lineKeys` parameter.
Assigning Static BLF and EFK to Line keys

If you're assigning static BLF and EFK to a line key, assign that line key to index = 1 to indicate the order in which they're defined.

Note: The parameter `linekey.x.index` must be configured to display static BLF and EFK when Flexible Line Key (FLK) is enabled.

Flexible Line Key Assignment Categories and Index

The FLK category specifies the function of the line key.

The index specifies the order in which the line keys display on the phone screen. Use the following table to help you assign a category and an index to the line keys on your phone. Note that the category Unassigned leaves the line key blank.

Flexible Line Key Assignment Categories and Index

<table>
<thead>
<tr>
<th>Category</th>
<th>Index</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unassigned</td>
<td>Null</td>
</tr>
<tr>
<td>Line</td>
<td>The Line index number.</td>
</tr>
<tr>
<td>BLF</td>
<td>Static BLF index number.</td>
</tr>
<tr>
<td>Speed Dial</td>
<td>The speed dial index number.</td>
</tr>
<tr>
<td>EFK</td>
<td>EFK index number.</td>
</tr>
<tr>
<td>Presence</td>
<td>0</td>
</tr>
</tbody>
</table>
Directories and Contacts

Topics:

• Local Contact Directory
• Speed Dials
• Corporate Directory
• Call Lists

You can configure phones with a local contact directory and link contacts to speed dial buttons. Additionally, call logs stored in the Missed Calls, Received Calls, and Placed Calls call lists let you view user phone events like remote party identification, time and date of call, and call duration. This section provides information on contact directory, speed dial, and call log parameters you can configure on your phone.

Local Contact Directory

Poly phones feature a contact directory file you can use to store frequently used contacts. The UC Software package includes a template contact directory file named 000000000000-directory~.xml that is loaded to the provisioning server the first time you boot up a phone with UC Software or when you reset the phone to factory default settings. When you first boot the phone out of the box or when you reset the phone to factory default settings, the phone looks for contact directories in the following order:

• An internally stored local directory
• A personal <MACaddress>-directory.xml file
• A global 000000000000-directory.xml file when the phone substitutes <000000000000> for its own MAC address.

In addition, make sure the dir.local.readonly parameter is enabled to restrict the users to modify speed dials.

Local Contact Directory Parameters

The following parameters configure the local contact directory.

**dir.local.contacts.maxNum**

Set the maximum number of contacts that can be stored in the Local Contact Directory. The maximum number varies by phone model, refer to section ‘Maximum Capacity of the Local Contact Directory’.

Change causes system to restart or reboot.

**dir.local.readonly**

0 (default) - Disable read only protection of the local Contact Directory.
1 - Enable read-only protection of the local Contact Directory.

**feature.directory.enabled**
0 - The local contact directory is disabled.
1 (default) - The local contact directory is enabled.

**dir.search.field**
Specify whether to sort contact directory searches by first name or last name.
0 (default) - Last name.
1 - First name.

**voIPProt.SIP.specialEvent.checkSync.downloadDirectory**
0 (default) - The phone downloads updated directory files after receiving a checksync NOTIFY message.
1 - The phone downloads the updated directory files along with any software and configuration updates after receiving a checksync NOTIFY message. The files are downloaded when the phone restarts, reboots, or when the phone downloads any software or configuration updates.

**Note:** The parameter `hotelingMode.type` set to 2 or 3 overrides this parameter.

**feature.pauseAndWaitDigitEntryControl.enabled**
1 (default) - Enable processing of control characters in the contact phone number field. When enabled, "," or "p" control characters cause a one second pause.

For example, "," or "p" control characters cause a one second pause. ";" or "w" control character cause a user prompt that allows a user-controlled wait. Subsequent digits entered to the contact field are dialed automatically.
0 - Disable processing of control characters.

**up.regOnPhone**
0 (default) - Contacts you assign to a line key display on the phone in the position assigned.
1 - Contacts you assign to a line key are pushed to the attached expansion module.
Change causes system to restart or reboot.

**Maximum Capacity of the Local Contact Directory**
The following table lists the maximum number of contacts and maximum file size of the local Contact Directory for each phone.

To conserve phone memory, use the parameter `dir.local.contacts.maxNum` to set a lower maximum number of contacts for the phones.
Maximum File Size and Number of Contacts

<table>
<thead>
<tr>
<th>Phone</th>
<th>Maximum File Size</th>
<th>Maximum Number of Contacts in File</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCX 400, 500, 600, and 700</td>
<td>4MB</td>
<td>500</td>
</tr>
</tbody>
</table>

Creating Per-Phone Directory Files

To create a per-phone, personal directory file, replace `<000000000000>` in the global file name with the phone's MAC address: `<MACaddress>-directory.xml`.

Any changes users make to the contact directory from the phone are stored on the phone drive and uploaded to the provisioning server in the personal directory `<MACaddress>-directory.xml` file, which enables you to preserve a contact directory during reboots.

To create a global directory file that you can use to maintain the directory for all phones from the provisioning server, remove the tilde (~) from the template file name `000000000000-directory.xml`. When you update the global directory file on the provisioning server, the updates are downloaded onto the phone and combined with the phone specific directory.

Maintaining Per-Phone Directory Files

Using the parameter `voIpProt.SIP.specialEvent.checkSync.downloadDirectory`, you can configure the phones to download updated directory files. The files are downloaded when the phone restarts, reboots, or when the phone downloads any software or configuration updates.

Any changes to either the global or personal directory files are reflected in the directory on the phone after a restarts. When merging the two files, the personal directory always takes precedence over the changes in the global directory. Thus, if a user modifies a contact from the global directory, the contact is saved in the personal directory file, and the contact from the global directory is ignored when the files are next uploaded.

The phone requests both the per-phone `<MACaddress>-directory.xml` and global contact directory `000000000000-directory.xml` files and merges them for presentation to the user. If you created a per-phone `<MACaddress>-directory.xml` for a phone, and you want to use the `000000000000-directory.xml` file, add the `000000000000-directory.xml` file to the provisioning server and update the phone's configuration.

Note: You can duplicate contacts in the Contact Directory on phones registered with the Ribbon Communications server.

Note: To avoid users accidentally deleting the definitions in the contact directory, make the contact directory file read only.

Speed Dials

You can link entries in the local contact directory to speed dial contacts to line keys on the Home or Lines screen to enable users to place calls quickly using dedicated speed dial buttons.

The number of supported speed dial entries varies by phone model.
### Speed Dial Index Ranges

<table>
<thead>
<tr>
<th>Phone Model</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCX 400, 500, 600, and 700</td>
<td>1 - 500</td>
</tr>
</tbody>
</table>

### Speed Dial Contacts Parameters

After setting up your per-phone directory file (`<MACaddress>-directory.xml`), enter a number in the speed dial `<sd>` field to display a contact directory entry as a speed dial contact on the phone. Speed dial entries automatically display on unused line keys on the phone and are assigned in numerical order.

On some call servers, enabling presence for an active speed dial contact displays that contact's status on the speed dial's line key label.

Use the parameter below, which identifies the directory XML file and the parameters you need to set up your speed dial contacts.

**dir.local.contacts.maxFavIx**

Configure the maximum number of speed dial contacts that can display on the Home screen.

Enter a speed dial index number in the `<sd>x</sd>` element in the `<MAC address>-directory.xml` file to display a contact directory entry as a speed dial key on the phone. Speed dial contacts are assigned to unused line keys and to entries in the phone's speed dial list in numerical order.

### Corporate Directory

You can connect phones to a corporate directory server that supports the Lightweight Directory Access Protocol (LDAP), version 3.

After you set up the corporate directory on the phones, users can search for contacts in the directory, place calls to directory contacts, and save entries to the local contact directory on the phone.

Poly phones support corporate directories that support server-side sorting and those that do not. For servers that do not support server-side sorting, sorting is performed on the phone.

**Note:** Use corporate directories that have server-side sorting for better performance. Consult your LDAP administrator when making any configuration changes for the corporate directory. For more information on LDAP attributes, see [RFC 4510 - Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map](https://tools.ietf.org/html/rfc4510).

### Securely Store LDAP Credentials

You can enable multiple users to enter their LDAP user credentials directly onto the phone to access the Corporate (LDAP) Directory, and you can enable the phones to store those credentials on the phone.

Any LDAP credentials entered on the phone are encrypted and stored on the phone only, and the credentials persist after the phone restarts or reboots.
When this feature is configured for phones with BroadSoft Flexible Seating, the phones can store up to 50 user credentials. If the number of user credentials reaches 50, the user who has the longest inactivity period is removed from the phone when any additional users are added.

**Procedure**

» Set the parameter `dir.corp.persistentCredentials` to **1**.

### Corporate Directory Parameters

Use the parameters in the following list to configure the corporate directory.

Note that the exact configuration of a corporate directory depends on the LDAP server you use.

---

**Note:** For detailed explanations and examples of all currently supported LDAP directories, see *Technical Bulletin 41137: Best Practices When Using Corporate Directory on Polycom Phones* at [Polycom Engineering Advisories and Technical Notifications](https://www.polycom.com/en-us/support/technical-advisories.html).

---

**dir.corp.address**

Set the IP address or hostname of the LDAP server interface to the corporate directory.

- Null (default)
- IP address
- Hostname
- FQDN

Change causes system to restart or reboot.

**dir.corp.allowCredentialsFromUI.enabled**

Enable or disable prompting users to enter LDAP credentials on the phone when accessing the Corporate Directory.

**Note:** Users are only prompted to enter their credentials when credentials are not added through configuration or after a login failure.

- 0 (default) - Disabled
- 1 - Enabled

**dir.corp.alt.transport**

Choose a transport protocol used to communicate to the corporate directory.

- TCP (default)
- TLS

**dir.corp.attribute.x.addstar**

Determine if the wild-card character, asterisk (*), is appended to the LDAP query field.

- 0 - Wild-card character is not appended.
- 1 (default) - Wild-card character is appended.
Change causes system to restart or reboot.

**dir.corp.attribute.x.filter**
Set the filter string for this parameter, which is edited when searching.
Null (default)
UTF-8 encoding string
Change causes system to restart or reboot.

**dir.corp.attribute.x.label**
Enter the label that shows when data is displayed.
Null (default)
UTF-8 encoding string
Change causes system to restart or reboot.

**dir.corp.attribute.x.name**
Enter the name of the parameter to match on the server. Each name must be unique; however, a global address book entry can have multiple parameters with the same name. You can configure up to eight parameters (x = 1 to 8).
Null (default)
UTF-8 encoding string
Change causes system to restart or reboot.

**dir.corp.attribute.x.searchable**
Determine whether quick search on parameter x (if x is 2 or more) is enabled or disabled.
0 (default) - Disabled
1 - Enabled
Change causes system to restart or reboot.

**dir.corp.attribute.x.sticky**
Sets whether the filter string criteria for attribute x is reset or retained after a phone reboot. If you set an attribute to be sticky (set this parameter to 1), a '*' displays before the label of the attribute on the phone.
0 (default) – Reset after a phone reboot.
1 – Retain after a phone reboot.
Change causes system to restart or reboot.

**dir.corp.attribute.x.type**
Define how x is interpreted by the phone. Entries can have multiple parameters of the same type. If the user saves the entry to the local contact directory on the phone, first_name, last_name, and phone_number are copied. The user can place a call to the phone_number and SIP_address from the global address book directory.
first_name
last_name (default)
phone_number
SIP_address
H323_address URL
other
Change causes system to restart or reboot.

dir.corp.auth.useLoginCredentials
0 (default) - Disabled
1 - Enabled

dir.corp.autoQuerySubmitTimeout
Set the timeout in seconds between when the user stops entering characters in the quick search
and when the search query is automatically submitted.
0 (default)
0 - 60
Change causes system to restart or reboot.

dir.corp.backGroundSync
Determine if background downloading from the LDAP server is enabled or disabled.
0 (default) - Disabled
1 - Enabled
Change causes system to restart or reboot.

dir.corp.backGroundSync.period
Set the time in seconds the corporate directory cache is refreshed after the corporate directory
feature has not been used for the specified period of time.
86400 (default)
3600 to 604800
Change causes system to restart or reboot.

dir.corp.baseDN
Enter the base domain name, which is the starting point for making queries on the LDAP server.
Null (default)
UTF-8 encoding string
Change causes system to restart or reboot.
**dir.corp.bindOnInit**
Enable or disabled use of bind authentication on initialization.
- 1 (default) - Enabled
- 0 - Disabled
Change causes system to restart or reboot.

**dir.corp.cacheSize**
Set the maximum number of entries that can be cached locally on the phone.
- 128 (default)
- 32 to 256
Change causes system to restart or reboot.

**dir.corp.customError**
Enter the error message to display on the phone when the LDAP server finds an error.
- Null (default)
- UTF-8 encoding string

**dir.corp.domain**
Enter the port that connects to the server if a full URL is not provided.
- 0 to 255

**dir.corp.filterPrefix**
Enter the predefined filter string for search queries.
- (objectclass=person) (default)
- UTF-8 encoding string
Change causes system to restart or reboot.

**dir.corp.pageSize**
Set the maximum number of entries requested from the corporate directory server with each query.
- 64 (default)
- 8 to 64
Change causes system to restart or reboot.

**dir.corp.password**
Enter the password used to authenticate to the LDAP server.
- Null (default)
- UTF-8 encoding string
**dir.corp.persistentCredentials**

Enable to securely store and encrypt LDAP directory user credentials on the phone. Enable `dir.corp.allowCredentialsFromUI.enabled` to allow users to enter credentials on the phone.

*Note:* If you disable the feature after enabling it, then all the saved user credentials are deleted.

0 (default) - Disabled

1 - Enabled

**dir.corp.port**

Enter the port that connects to the server if a full URL is not provided.

389 (default for TCP)

636 (default for TLS)

0

Null

1 to 65535

Change causes system to restart or reboot.

**dir.corp.querySupportedControlOnInit**

Enable to make the phone make an initial query to check the status of the server when booting up.

0 - Disabled

1 (default) - Enabled

**dir.corp.scope**

sub (default) - a recursive search of all levels below the base domain name is performed.

one - a search of one level below the base domain name is performed.

base - a search at the base domain name level is performed.

Change causes system to restart or reboot.

**dir.corp.serverSortNotSupported**

0 (default) - The server supports server-side sorting.

1 - The server does not support server-side sorting, so the phone handles the sorting.

**dir.corp.sortControl**

Determine how a client can make queries and sort entries.

0 (default) - Leave sorting as negotiated between the client and server.

1 - Force sorting of queries, which causes excessive LDAP queries and should only be used to diagnose LDAP servers with sorting problems.
Change causes system to restart or reboot.

**dir.corp.transport**
Specify whether a TCP or TLS connection is made with the server if a full URL is not provided.
- TCP (default)
- TLS
- Null
Change causes system to restart or reboot.

**dir.corp.user**
Enter the user name used to authenticate to the LDAP server.
- Null (default)
- UTF-8 encoding string

**dir.corp.viewPersistence**
- 0 (default) - The corporate directory search filters and browsing position are reset each time the user accesses the corporate directory.
- 1 - The search filters and browsing position from the previous session are displayed each time the user accesses the corporate directory.
Change causes system to restart or reboot.

**dir.corp.vlv.allow**
Determine whether virtual view list (VLV) queries are enabled and can be made if the LDAP server supports VLV.
- 0 (default)
- 1
Change causes system to restart or reboot.

**dir.corp.vlv.sortOrder**
Enter the list of parameters, in exact order, for the LDAP server to use when indexing. For example: `sn, givenName, telephoneNumber`.
- Null (default)
- list of parameters
Change causes system to restart or reboot.

**feature.contacts.enabled**
- 1 (default) - The Contacts icon displays on the Home screen, the global menu, and in the dialer.
- 0 - Disable display of the Contacts icon.
**feature.corporateDirectory.enabled**

0 (default) - The corporate directory feature is disabled and the icon is hidden.
1 (default) - The corporate directory is enabled and the icon shows.

**Call Lists**

The phone records and maintains user phone events to a call list, which contains call information such as remote party identification, time and date of the call, and call duration.

The list is stored on the provisioning server as an XML file named `<MACaddress>-calls.xml`. If you want to route the call list to another server, use the `CALL_LISTS_DIRECTORY` field in the master configuration file. All call lists are enabled by default.

The phone maintains all the calls in three separate user accessible call lists: Missed Calls, Received Calls, and Placed Calls. Users can clear lists manually on their phones, or delete individual records or all records in a group (for example, all missed calls).

**Call List Parameters**

Use the following parameters to configure call lists.

**callLists.collapseDuplicates**

Generic Base Profile - 1 (default)

1 - Consecutive incomplete calls to/from the same party and in the same direction are collapsed into one record in the calls list. The collapsed entry displays the number of consecutive calls.

0 - Each call is listed individually in the calls list.

**callLists.logConsultationCalls**

Generic Base Profile - 1 (default)

0 - Consultation calls not joined into a conference call aren't listed as separate calls in the calls list.

1 - Each consultation call is listed individually in the calls list.

**feature.callList.enabled**

1 (default) - Allows you to enable the missed, placed, and received call lists on all phone menus including the Home screen and dial pad.

0 - Disables all call lists.

**feature.callListMissed.enabled**

0 (Default) - The missed call list is disabled.

1 - The missed call list is enabled.

To enable the missed, placed, or received call lists, `feature.callList.enabled` must be enabled.
**feature.callListPlaced.enabled**

0 (Default) - The placed call list is disabled.
1 - The placed call list is enabled.

To enable the missed, placed, or received call lists, `feature.callList.enabled` must be enabled.

**feature.callListReceived.enabled**

0 (Default) - The received call list is disabled.
1 - The received call list is enabled.

To enable the missed, placed, or received call lists, `feature.callList.enabled` must be enabled.

**feature.exchangeCallLog.enabled**

If Base Profile is:

- Generic - 0 (default)

1 - The Exchange call log feature is enabled, user call logs are synchronized with the server, and the user call history of Missed, Received, and outgoing calls can be retrieved on the phone.

You must also enable the parameter `feature.callList.enabled` to use the Exchange call log feature.

0 - The Exchange call log feature is disabled, the user call log history can’t be retrieved from the Exchange server, and the phone generates call logs locally.

**Call Log Elements and Attributes**

The following table describes each element and attribute that displays in the call log.

You can place the elements and attributes in any order in your configuration file.

**Call Log Elements and Attributes**

<table>
<thead>
<tr>
<th>Element</th>
<th>Permitted Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>direction</td>
<td>In, Out</td>
</tr>
<tr>
<td>line</td>
<td>Positive integer</td>
</tr>
<tr>
<td>protocol</td>
<td>SIP or H323</td>
</tr>
<tr>
<td>disposition</td>
<td>Busy, Forwarded, Normal, Partial, Preempted, Rejected, RemotelyHandled, Transferred</td>
</tr>
</tbody>
</table>

Call direction with respect to the user.

The line (or registration) index.

The line protocol.
<table>
<thead>
<tr>
<th>Element</th>
<th>Permitted Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>startTime</td>
<td>String</td>
</tr>
<tr>
<td></td>
<td>The start time of the call. For example: 2010-01-05T12:38:05 in local time.</td>
</tr>
<tr>
<td>duration</td>
<td>String</td>
</tr>
<tr>
<td></td>
<td>The duration of the call, beginning when it is connected and ending when the call is terminated. For example: PT1H10M59S.</td>
</tr>
<tr>
<td>count</td>
<td>Positive Integer</td>
</tr>
<tr>
<td></td>
<td>The number of consecutive missed and abandoned calls from a call destination.</td>
</tr>
<tr>
<td>destination</td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td>The original destination of the call. For outgoing calls, this parameter designates the outgoing call destination; the name is initially supplied by the local phone (from the name field of a local contact entry) but may later be updated via call signaling. This field should be used for basic redial scenarios. For incoming calls, the called destination identifies the requested party, which may be different than any of the parties that are eventually connected (the destination may indicate a SIP URI which is different from any SIP URI assigned to any lines on the phone).</td>
</tr>
<tr>
<td>source</td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td>The source of the call (caller ID from the call recipient's perspective).</td>
</tr>
<tr>
<td>Connection</td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td>An array of connected parties in chronological order. As a call progresses, the connected party at the far end may change, for example, if the far end transfers the call to someone else. The connected element allows the progression of connected parties, when known, to be saved for later use. All calls that contain a connected state must have at least one connection element created.</td>
</tr>
<tr>
<td>finalDestination</td>
<td>Address</td>
</tr>
<tr>
<td></td>
<td>The final connected party of a call that has been forwarded or transferred to a third party.</td>
</tr>
</tbody>
</table>
Call Controls

Topics:

- Microphone Mute
- Persistent Microphone Mute
- Call Timers
- Called Party Identification
- Connected Party Identification
- Calling Party Identification
- Remote Party Caller ID from SIP Messages
- Connected Line Identification
- Calling Line Identification
- SIP Header Warnings
- Accessing URLs in SIP Messages
- Distinctive Incoming Call Treatment
- Distinctive Call Waiting
- Presence Status
- Do Not Disturb
- Remote Party Disconnect Alert Tone
- Call Waiting Alerts
- Missed Call Notifications
- Last Call Return
- Call Hold
- Call Hold Timer
- Call Park and Retrieve
- Call Transfer
- Call Forwarding
- Automatic Off-Hook Call Placement
- Directed Call Pickup
- Group Call Pickup
- Multiple Line Registrations
- Multiple Line Keys Per Registration
- Multiple Call Appearances
- Flexible Call Appearances
This chapter shows you how to configure call control features.

Microphone Mute

All phones have a microphone mute button.

By default, when you activate microphone mute, a red LED glows or a mute icon displays on the phone screen, depending on the phone model you are using.

You cannot configure the microphone mute feature.

Persistent Microphone Mute

With this feature, you can enable the microphone mute to persist across all calls managed on a phone.

By default, users can mute the microphone during an active call and it is unmuted when the active call ends. With persistent microphone mute enabled, when a user mutes the microphone during an active call, the microphone remains muted for all following calls until the user unmutes the microphone or the phone restarts.

When a user mutes the microphone when the phone is idle, the mute LED glows but no icon displays on the screen. When a user initiates a new active call with the microphone muted, the mute LED glows and a Mute icon displays on the phone screen.

Persistent Microphone Mute Parameter

Use the following parameter to enable persistent microphone mute.

```
feature.persistentMute.enabled
```

0 - The mute state ends when the active call ends or when the phone restarts.
1 - When a user mutes the microphone during an active call, the microphone remains muted for all following calls until the user unmutes the microphone or the phone restarts.
Change causes system to restart or reboot.

Call Timers
By default, a call timer displays on the phone’s screen during calls, and a separate call duration timer displays the hours, minutes, and seconds for each call in progress.
You can’t configure the call timer display.

Called Party Identification
By default, the phone displays and logs the identity of all parties on outgoing calls.
The phone obtains called party identities from network signaling. Because party identification on outgoing calls is a default feature, the phone displays caller IDs matched to the call server and does not match IDs to entries in the contact directory or corporate directory.

Calling Party Identification Parameters
Use the parameters in the following list to configure calling party identification.

call.callsPerLineKey
Set the maximum number of concurrent calls per line key. This parameter applies to all registered lines and can be overridden by the per-registration parameter reg.x.callsPerLineKey.
24 (default)
1 - 24

up.useDirectoryNames
1 (default) - The name field in the local contact directory is used as the caller ID for incoming calls from contacts in the local directory. Note: Outgoing calls and corporate directory entries are not matched.
0 - Names provided through network signaling are used for caller ID.

Connected Party Identification
By default, the phone displays and logs the identities of remote parties you connect to if the call server can derive the name and ID from network signaling.
In cases where remote parties have set up certain call features, the remote party you connect to—and the caller ID that displays on the phone—may be different than the intended party's. For example, Bob places a call to Alice, but Alice has call diversion configured to divert Bob's incoming calls to Fred. In this case, the phone logs and displays the connection between Bob and Fred. The phone does not match party IDs to entries in the contact directory or the corporate directory.
Calling Party Identification

By default, the phone displays the identity of incoming callers if available to the phone through the network signal.

If the incoming call address has been assigned to the contact directory, you can enable the phones to display the name assigned to contacts in the contact directory. However, the phone cannot match the identity of calling parties to entries in the corporate directory.

Calling Party Identification Parameters

Use the parameters in the following list to configure Calling Party Identification.

\[ \text{up.useDirectoryNames} \]

1 (default) - The name field in the local contact directory is used as the caller ID for incoming calls from contacts in the local directory. Note: Outgoing calls and corporate directory entries are not matched.

0 - Names provided through network signaling are used for caller ID.

Remote Party Caller ID from SIP Messages

You can specify which SIP request and response messages to use to retrieve caller ID information.

Remote Party Caller ID from SIP Messages Parameters

Use the following parameters to specify which SIP request and response messages to use to retrieve caller ID information.

\[ \text{voIpProt.SIP.CID.request.sourceSipMessage} \]

Specify which header in the SIP request to retrieve remote party caller ID from. You can use:

- \text{voIpProt.SIP.callee.sourcePreference} 
- \text{voIpProt.SIP.caller.sourcePreference} 
- \text{voIpProt.SIP.CID.sourcePreference}

UPDATE takes precedence over the value of this parameter.

NULL (default) - Remote party caller ID information from INVITE is used.

INVITE
PRACK
ACK
0-6

This parameter does not apply to shared lines.
**voIpProt.SIP.CID.response.sourceSipMessage**

Specify which header in the SIP request to retrieve remote party caller ID from. You can use:

- voIpProt.SIP.callee.sourcePreference
- voIpProt.SIP.caller.sourcePreference
- voIpProt.SIP.CID.sourcePreference

NULL (default) - The remote party caller ID information from the last SIP response is used.

100, 180, 183, 200

0-3

This parameter does not apply to shared lines.

---

**Connected Line Identification**

You can view the identity of the callee on the caller's phone screen.

If the contact details are stored on your phone, the saved contact name and number will be displayed.

---

**Calling Line Identification**

The Calling Line Identity Presentation (CLIP) displays the phone number of the caller on the phone screen.

You can configure this feature by using the parameters in the following table.

---

**Calling Line Identification Parameters**

**voIpProt.SIP.CID.sourcePreference**

Specify the priority order for the sources of caller ID information. The headers can be in any order.

Null (default) - Caller ID information comes from P-Asserted-Identity, Remote-Party-ID, and From in that order.

From,P-Asserted-Identity, Remote-Party-ID

Supported Headers Default Order: P-Asserted-Identity,Remote-Party-ID,From

**Note:** By default callee and caller will take identity order from voIpProt.SIP.CID.sourcePreference.

If voIpProt.SIP.Caller.SourcePreference or voIpProt.SIP.Callee.SourcePreference are configured then the order set by voIpProt.SIP.CID.sourcePreference is ignored.
**voIpProt.SIP.caller.sourcePreference**

Set the priority order to display the caller's identity for incoming calls.

- Null (default)
- 0-120

**Supported Headers Default Order:** P-Asserted-Identity, Remote-Party-ID, From

**String**

**voIpProt.SIP.callee.sourcePreference**

Set the priority order to display the callee's identity for outgoing calls.

- Null (default)
- Supported Headers Default Order: P-Asserted-Identity, Remote-Party-ID, From

**String**

**SIP Header Warnings**

You can configure the warning field from a SIP header to display a pop-up message on the phone, for example, when a call transfer failed due to an invalid extension number.

You can display pop-up messages in any language supported by the phone. The messages display for three seconds unless overridden by another message or action.

For a list of supported SIP header warnings, see the article "Supported SIP Request Headers" in the Polycom Knowledge Base.

**SIP Header Warning Parameters**

You can use the parameters in the following list to enable the warning display or specify which warnings to display.

**voIpProt.SIP.header.warning.enable**

- 0 (default) - The warning header is not displayed.
- 1 - The warning header is displayed if received.

**voIpProt.SIP.header.warning.codes.accept**

Specify a list of accepted warning codes.

- Null (default) - All codes are accepted. Only codes between 300 and 399 are supported.

For example, if you want to accept only codes 325 to 330:

```
voIpProt.SIP.header.warning.codes.accept=325,326,327,328,329,330
```
Accessing URLs in SIP Messages

When this feature is enabled, the server attaches a URL to incoming and active calls.

The web browser or microbrowser can read this URL and present it as web content that displays on the phone screen.

This feature is flexible and can be used in some of the following ways:

- In a Call Center environment, the phone displays extended information about a customer before the agent takes the call. The phone can also display a script of questions for the agent to ask during the call.
- In a hotel, a guest can view the restaurant menu on the phone.

Access URL in SIP Messages Parameters

You can configure the retrieval method for web content and enable users to choose to retrieve web content using either Active or Passive mode.

If your call server supports access URLs, you can also specify active or passive retrieval in the SIP header. If parameters in the SIP signal conflict with the file configuration, parameters in the SIP signaling take precedence.

You can also enable new web content to be added to the Settings menu on the phone, and users can set the default display mode for individual URLs to active or passive from the phone’s menu.

mb.ssawc.enabled

- 0 (default) - Spontaneous display of web content is disabled.
- 1 - Spontaneous web content display is enabled.

mb.ssawc.call.mode

- passive (default) - Web content is displayed only when requested by the user. Passive mode is recommended when the microbrowser is used for other applications. When passive mode is enabled, an icon displays beside a call appearance indicating that web content is available, and the user can press Select to view the content.
- Active - Web content is retrieved spontaneously and displayed immediately.

Distinctive Incoming Call Treatment

You can apply distinctive treatment to specific calls and contacts in the contact directory.

You can set up distinctive treatment for each of your contacts by specifying a Divert Contact, enabling Auto-Reject, or enabling Auto-Divert for a specific contact in the local contact directory. You can also apply distinctive treatment to calls and contacts through the phone’s user interface.

If you enable both the auto divert and auto reject features, auto divert has precedence over auto reject.
Distinctive Call Waiting

You can use the alert-info values and class fields in the SIP header to map calls to distinct call-waiting types.

You can apply three call waiting types: beep, ring, and silent. This feature requires call server support.

Distinctive Call Waiting Parameters

You can use the alert-info values and class fields in the SIP header to map calls to distinct call-waiting types.

\[ \text{voIpProt.SIP.alertInfo.x.class} \]

Alert-Info fields from INVITE requests are compared as many of these parameters as are specified (x=1, 2, ..., N) and if a match is found, the behavior described in the corresponding ring class is applied.

default (default)

\[ \text{voIpProt.SIP.alertInfo.x.value} \]

Specify a ringtone for single registered line using a string to match the Alert-Info header in the incoming INVITE.

NULL (default)

Presence Status

You can enable users to monitor the status of other remote users and phones.

By adding remote users to a buddy list, users can monitor changes in the status of remote users in real time or they can monitor remote users as speed-dial contacts. Users can also manually specify their status in order to override or mask automatic status updates to others and can receive notifications when the status of a remote line changes.

Poly phones support a maximum of:

- 64 buddies for Open SIP server platforms

Presence Status Parameters

Use the following parameters to enable Presence and display the MyStatus and Buddies soft keys on the phone.

\[ \text{feature.presence.enabled} \]

0 (default) - Disable the presence feature—including buddy managements and user status.
1 - Enable the presence feature with the buddy and status options.
**pres.idleSoftkeys**

- 1 (default) - The MyStat and Buddies presence idle soft keys display.
- 0 - The MyStat and Buddies presence idle soft keys do not display.

**pres.reg**

The valid line/registration number to use for presence. If the value is not a valid registration, this parameter is ignored.

1 (default)
1 - 34

---

**Do Not Disturb**

You can enable Do Not Disturb (DND) locally on the phone or on the server.

The local DND feature is enabled by default, and users can enable or disable DND for all or individual registered lines on the phone. When enabled, users are not notified of incoming calls placed to their line.

**Server-Based Do Not Disturb**

If you want to enable server-based DND, you must enable the feature on both a registered phone and on the server.

The following conditions apply for server-based DND:

- Server-based DND can be applied to multiple registered lines on a phone; however, applying DND to individual registrations is not supported.
- Server-based DND cannot be enabled on a phone configured as a shared line.
- If server-based DND is enabled but not turned on when the DND feature is enabled on the phone, the "Do Not Disturb" message displays on the phone, but incoming calls continue to ring.
- Server-based DND disables local Call Forward and DND, however, if an incoming is not routed through the server, an audio alert still plays on the phone.

**Do Not Disturb Parameters**

Use the parameters in the following list to configure the local DND feature.

**feature.doNotDisturb.enable**

1 (default) - Enable Do Not Disturb (DND).
0 - Disable Do Not Disturb (DND).

Change causes system to restart or reboot.

**voIpProt.SIP.serverFeatureControl.dnd**

0 (default) - Disable server-based DND.
1 - Server-based DND is enabled. Server and local phone DND are synchronized.
**voIpProt.SIP.serverFeatureControl.localProcessing.dnd**

This parameter depends on the value of `voIpProt.SIP.serverFeatureControl.dnd`. If set to 1 (default) and `voIpProt.SIP.serverFeatureControl.dnd` is set to 1, the phone and the server perform DND.

If set to 0 and `voIpProt.SIP.serverFeatureControl.dnd` is set to 1, DND is performed on the server-side only, and the phone does not perform local DND.

If both `voIpProt.SIP.serverFeatureControl.localProcessing.dnd` and `voIpProt.SIP.serverFeatureControl.dnd` are set to 0, the phone performs local DND and the `localProcessing` parameter is not used.

1 (default) - Enabled
0 - Disabled

**call.rejectBusyOnDnd**

When enabled, the phone rejects incoming calls with a busy signal while Do Not Disturb is on. When disabled, the phone gives a visual alert of incoming calls, but no audible ring, when Do Not Disturb is on.

1 (default) - Enabled
0 - Disabled

*Note:* This parameter does not apply to shared lines since not all users may want DND enabled.

Change causes system to restart or reboot.

**call.donotdisturb.perReg**

This parameter determines if the do-not-disturb feature applies to all registrations on the phone or on a per-registration basis.

0 (default) - DND applies to all registrations on the phone.
1 - Users can activate DND on a per-registration basis.

*Note:* If `voIpProt.SIP.serverFeatureControl.dnd` is set to 1 (enabled), this parameter is ignored.

---

**Remote Party Disconnect Alert Tone**

Remote Party Disconnect Alert Tone alerts users when the call has been disconnected by a remote party or network.

When a remote party or network on an active call gets disconnected, an alert is played to notify the user about the lost connection. The tone is played only for an active call.
Remote Party Disconnect Alert Tone Parameter

You can configure this feature by using the parameter below.

call.remoteDisconnect.toneType

Choose an alert tone to play when the remote party disconnects call.

Silent (Default)

messageWaiting, instantMessage, remoteHoldNotification, localHoldNotification,
positiveConfirm, negativeConfirm, welcome, misc1, misc2, misc3, misc4, misc5, misc6, misc7,
custom1, custom2, custom3, custom4, custom5, custom6, custom7, custom8, custom9,
custom10

Call Waiting Alerts

By default, the phone alerts users to incoming calls while a user is in an active call.

You can choose to disable these call waiting alerts and specify ringtones for incoming calls.

Call Waiting Alert Parameters

Use the parameters in the following list to configure call waiting alerts.

call.callWaiting.enable

Enable or disable call waiting.

1 (default) - The phone alerts you to an incoming call while you are in an active call. If 1, and
you end the active call during a second incoming call, you are alerted to the second incoming
call.

0 - You are not alerted to incoming calls while in an active call. The incoming call is treated as if
you did not answer it.

call.callWaiting.ring

Specifies the ringtone of incoming calls when another call is active. If no value is set, the default
value is used.

beep (default) - A beep tone plays through the selected audio output mode on the active call.
ring - The configured ringtone plays on the speaker.
silent - No ringtone.

Missed Call Notifications

By default, a counter with the number of missed calls displays on the Recent Calls icon on the phone.

You can configure the phone to record all missed calls or to display only missed calls that arrive through
the SIP server. You can also enable missed call notifications for each registered line on a phone.
Missed Call Notification Parameters

Use the following list to configure options for missed call notifications.

**call.missedCallTracking.x.enabled**
- 1 (default) - Missed call tracking for a specific registration is enabled.
- If `call.missedCallTracking.x.enabled` is set to 0, then the missed call counter is not updated regardless of what `call.serverMissedCalls.x.enabled` is set to (and regardless of how the server is configured) and the missed call list does not display in the phone menu.
- If `call.missedCallTracking.x.enabled` is set to 1 and `call.serverMissedCalls.x.enabled` is set to 0, then the number of missed calls is incremented regardless of how the server is configured.
- If `call.missedCallTracking.x.enabled` is set to 1 and `call.serverMissedCalls.x.enabled` is set to 1, then the handling of missed calls depends on how the server is configured.

Change causes system to restart or reboot.

**call.serverMissedCall.x.enabled**
- 0 (default) - All missed-call events increment the counter for a specific registration.
- 1 - Only missed-call events sent by the server will increment the counter.

**Note:** This feature is supported only with the BroadSoft Synergy call server (previously known as Sylantro).

Change causes system to restart or reboot.

Last Call Return

The phone supports redialing the last received call.

This feature requires support from a SIP server. With many SIP servers, this feature is implemented using a particular star code sequence. With some SIP servers, specific network signaling is used to implement this feature. When enabled, the phone displays an LCR soft key that users can select to place a call to the phone address that last called them.

Last Call Return Parameters

The last call return string value that you enter for parameter `call.lastCallReturnString` depends on the call server you use. Consult with your call server provider for the last call return string.

**feature.lastCallReturn.enabled**
- 0 (default) - Disable last call return feature.
- 1 - Enable last call return.
call.lastCallReturnString
  Specify the string sent to the server when the user selects the last call return action. The string
  is usually a star code.
  *69 (default)
  string - maximum 32 characters

Call Hold
Call hold enables users to pause activity on an active call so that they can use the phone for another task,
such as searching the phone's menu for information.
When an active call is placed on hold, a message displays informing the held party that they are on hold.
If supported by the call server, you can enter a music-on-hold URI. For more information, see RFC Music
on Hold draft-worley-service-example.

Call Hold Parameters
See the following list for the available parameters you can use to configure for Call Hold.

voIpProt.SIP.useRFC2543hold
  0 (default) - SDP media direction parameters (such as a=sendonly) per RFC 3264 when
  initiating a call.
  1 - the obsolete c=0.0.0.0 RFC2543 technique is used when initiating a call.

voIpProt.SIP.useSendonlyHold
  1 (default) - The phone will send a reinvite with a stream mode parameter of "sendonly" when
  a call is put on hold.
  0 - The phone will send a reinvite with a stream mode parameter of "inactive" when a call is
  put on hold

  Note: The phone will ignore the value of this parameter if set to 1 when the parameter
  voIpProt.SIP.useRFC2543hold is also set to 1 (default is 0).

call.hold.localReminder.enabled
  0 (default) - Users are not reminded of calls that have been on hold for an extended period of
  time.
  1 - Users are reminded of calls that have been on hold for an extended period of time.
  Change causes system to restart or reboot.

call.hold.localReminder.period
  Specify the time in seconds between subsequent hold reminders.
  60 (default)
Change causes system to restart or reboot.

**call.hold.localReminder.startDelay**

Specify a time in seconds to wait before the initial hold reminder.

90 (default)

Change causes system to restart or reboot.

**voIpProt.SIP.musicOnHold.uri**

A URI that provides the media stream to play for the remote party on hold. This parameter is used if `reg.x.musicOnHold.uri` is Null.

Null (default)

SIP URI

**Hold Implementation**

Poly phones support two currently accepted means of signaling hold, and you can configure phones to use either hold signaling method.

Poly phones support both methods when signaled by a remote endpoint.

**Supported Hold Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal the media directions with the &quot;a&quot; SDP media attributes sendonly, recvonly, inactive, or sendrecv.</td>
<td>Preferred method.</td>
</tr>
<tr>
<td>Set the &quot;c&quot; destination addresses for the zmedia streams in the SDP to zero. For example, c=0.0.0.0</td>
<td>No longer recommended due to RTCP problems associated with this method. Receiving sendrecv, sendonly, or inactive from the server causes the phone to revert to the other hold method.</td>
</tr>
</tbody>
</table>

**Call Hold Timer**

Poly phones display the timer when an active call is put on hold. The timer shows how long a call has been on hold.

**Call Hold Timer Parameter**

Use the following parameter to configure Call Hold Timer.

**up.holdTimerDisplay.enable**

0 (default) – Hold Timer will not display.

1 – Hold Timer will display.
**up.timerDisplayInSeconds**

0 (default) – The call timer and call hold timer are displayed in “hh:mm:ss” notation.
1 – Call timer is displayed in 5-digit second notation as “sssss” notations, and the call hold timer is displayed in 4-digit second notation as “ssss” notations.

---

**Call Park and Retrieve**

This feature enables users to park an active call to a call orbit and retrieve parked calls from the call orbit on any phone.

Whereas call hold keeps the held call on the same line, call park moves the call to a separate address where the call can be retrieved by any phone. This feature requires support from a SIP server and setup of this feature depends on the SIP server. For example, while some SIP servers implement group call pick-up using a particular star-code sequence, others implement the feature using network signaling.

You can also restrict the user to park an active call to a park orbit which already has a call parked. You can configure this feature using configuration parameter.

---

**Call Park and Retrieve Parameters**

Use the parameter below to configure Call Park and Retrieve.

**call.parkedCallString**

The star code to initiate the call park.

- **String**
- *68 (default)*

Change causes system to restart or reboot.

---

**Call Transfer**

The call transfer feature enables users to transfer an existing active call to a third-party address. You can configure the call transfer feature and set the default transfer type.

Users can perform the following types of call transfers:

- **Blind Transfer**—Users complete a call transfer without speaking with the other party first.
- **Consultative Transfer**—Users speak with the other party before completing the transfer.

By default, users can complete a call transfer without waiting for the other party to answer the call first, which is a Blind Transfer. In this case, Party A can transfer Party B’s call to Party C before Party C answers the transferred call. You can disable the blind transfer feature so that users must wait for the other party to answer before completing the transfer.
Call Transfer Parameters

Use the following list to specify call transfer behavior.

**voIpProt.SIP.allowTransferOnProceeding**

1 (default) - Transfer during the proceeding state of a consultation call is enabled.
0 - Transfer during the proceeding state of a consultation call is disabled
2 - Phones will accept an INVITE with replaces for a dialog in early state. This is needed when using transfer on proceeding with a proxy call server such as openSIPS, reSIProcate or SipXecs.

**call.defaultTransferType**

Set the transfer type the phone uses when transferring a call.

Generic Base Profile: Consultative (default) - Users can immediately transfer the call to another party.

Call Forwarding

Poly phones support a flexible call forwarding feature that enables users to forward incoming calls to another contact or phone line.

Users can enable call forwarding in the following ways:

- To all calls
- To incoming calls from a specific caller or extension
- During an incoming call
- When the phone is busy
- When do not disturb is enabled
- After a set number of rings before the call is answered
- To a predefined destination chosen by the user

Call Forward on Shared Lines

You can enable server-based call forwarding for shared lines.

If using BroadWorks R20 server, note the following:

- Local call-forwarding is not supported on shared lines.
- Dynamic call forwarding—forwarding incoming calls without answering the call—is not supported.

**Note:** The server-based and local call forwarding features do not work with the shared call appearance (SCA) and bridged line appearance (BLA) features. In order to enable users to use call forwarding, disable SCA or BLA enabled.
Call Forwarding Parameters

Use the parameters in the following list to configure feature options for call forwarding.

**feature.forward.enable**

1 (default) - Enables call forwarding.
0 - Disables call forwarding. Users cannot use Call Forward and the option is removed from the phone’s Features menu.

**voIPProt.SIP.serverFeatureControl.cf**

0 (default) - The server-based call forwarding is not enabled.
1 - The server-based call forwarding is enabled.
Change causes system to restart or reboot.

**voIPProt.SIP.serverFeatureControl.localProcessing.cf**

This parameter depends on the value of voIPProt.SIP.serverFeatureControl.cf.

1 (default) - If set to 1 and voIPProt.SIP.serverFeatureControl.cf is set to 1, the phone and the server perform call forwarding.
0 - If set to 0 and voIPProt.SIP.serverFeatureControl.cf is set to 1, call forwarding is performed on the server side only, and the phone does not perform local call forwarding.
If both voIPProt.SIP.serverFeatureControl.localProcessing.cf and voIPProt.SIP.serverFeatureControl.cf are set to 0, the phone performs local call forwarding and the localProcessing parameter is not used.

**voIPProt.SIP.header.diversion.enable**

0 (default) - If set to 0, the diversion header is not displayed.
1 - If set to 1, the diversion header is displayed if received.
Change causes system to restart or reboot.

**voIPProt.SIP.header.diversion.list.useFirst**

1 (default) - If set to 1, the first diversion header is displayed.
0 - If set to 0, the last diversion header is displayed.
Change causes system to restart or reboot.

**divert.x.contact**

All automatic call diversion features uses this forward-to contact. All automatically forwarded calls are directed to this contact. The contact can be overridden by a busy contact, DND contact, or no-answer contact as specified by the busy, dnd, and noAnswer parameters that follow.

Null (default)

string - Contact address that includes ASCII encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL (6416 or 6416@polycom.com).
Change causes system to restart or reboot.

**divert.x.sharedDisabled**
- 1 (default) - Disables call diversion features on shared lines.
- 0 - Enables call diversion features on shared lines.
Change causes system to restart or reboot.

**divert.x.autoOnSpecificCaller**
- 1 (default) - Enables the auto divert feature of the contact directory for calls on registration x. You can specify to divert individual calls or divert all calls.
- 0 - Disables the auto divert feature of the contact directory for registration x.
Change causes system to restart or reboot.

**divert.busy.x.enabled**
- 1 (default) - Diverts calls registration x is busy.
- 0 - Does not divert calls if the line is busy.
Change causes system to restart or reboot.

**divert.busy.x.contact**
Calls are sent to the busy contact's address if it is specified; otherwise calls are sent to the default contact specified by `divert.x.contact`.
- Null (default) string - contact address.
Change causes system to restart or reboot.

**divert.dnd.x.enabled**
- 0 (default) - Divert calls when DND is enabled on registration x.
- 1 - Does not divert calls when DND is enabled on registration x.
Change causes system to restart or reboot.

**divert.dnd.x.contact**
Calls are sent to the DND contact's address if it is specified; otherwise calls are sent to the default contact specified by `divert.x.contact`.
- Null (default)
- string - contact address.
Change causes system to restart or reboot.

**divert.fwd.x.enabled**
- 1 (default) - Users can forward calls on the phone's Home screen and use universal call forwarding.
0 - Users cannot enable universal call forwarding (automatic forwarding for all calls on registration x).
Change causes system to restart or reboot.

**divert.noanswer.x.enabled**

1 (default) - Unanswered calls after the number of seconds specified by timeout are sent to the no-answer contact.
0 - Unanswered calls are diverted if they are not answered.
Change causes system to restart or reboot.

**divert.noanswer.x.contact**

Null (default) - The call is sent to the default contact specified by divert.x.contact.
string - contact address
Change causes system to restart or reboot.

**divert.noanswer.x.timeout**

55 (default) - Number of seconds for timeout.
positive integer
Change causes system to restart or reboot.

**reg.x.fwd.busy.contact**

The forward-to contact for calls forwarded due to busy status.
Null (default) - The contact specified by divert.x.contact is used.
string - The contact specified by divert.x.contact is not used

**reg.x.fwd.busy.status**

0 (default) - Incoming calls that receive a busy signal is not forwarded
1 - Busy calls are forwarded to the contact specified by reg.x.fwd.busy.contact.

**reg.x.fwd.noanswer.contact**

Null (default) - The forward-to contact specified by divert.x.contact is used.
string - The forward to contact used for calls forwarded due to no answer.

**reg.x.fwd.noanswer.ringCount**

The number of seconds the phone should ring for before the call is forwarded because of no answer. The maximum value accepted by some call servers is 20.
0 - (default)
1 to 65535
**reg.x.fwd.noanswer.status**

- **0 (default)** - The calls are not forwarded if there is no answer.
- **1** - The calls are forwarded to the contact specified by `reg.x.noanswer.contact` after ringing for the length of time specified by `reg.x.fwd.noanswer.ringCount`.

**reg.x.serverFeatureControl.cf**

This parameter overrides `voIpProt.SIP.serverFeatureControl.cf`.

- **0 (default)** - The server-based call forwarding is disabled.
- **1** - Server-based call forwarding is enabled.

Change causes system to restart or reboot.

**divert.x.sharedDisabled**

- **1 (default)** - Disables call diversion features on shared lines.
- **0** - Enables call diversion features on shared lines.

Change causes system to restart or reboot.

**voIpProt.SIP.serverFeatureControl.cf**

- **0 (default)** - Disable server-based call forwarding.
- **1** - Enable server-based call forwarding.

This parameter overrides `reg.x.serverFeatureControl.cf`.

Change causes system to restart or reboot.

**voIpProt.SIP.serverFeatureControl.localProcessing.cf**

- **1 (default)** - Allows to use the value for `voIpProt.SIP.serverFeatureControl.cf`.
- **0** - Does not use the value for

This parameter depends on the value of `voIpProt.SIP.serverFeatureControl.cf`.

**reg.x.serverFeatureControl.localProcessing.cf**

This parameter overrides `voIpProt.SIP.serverFeatureControl.localProcessing.cf`.

- **0 (default)** - If `reg.x.serverFeatureControl.cf` is set to 1 the phone does not perform local Call Forward behavior.
- **1** - The phone performs local Call Forward behavior on all calls received.

**call.shared.disableDivert**

- **1 (default)** - Enable the diversion feature for shared lines.
- **0** - Disable the diversion feature for shared lines. Note that this feature is disabled on most call servers.

Change causes system to restart or reboot.
Automatic Off-Hook Call Placement

You can configure the phone to automatically place a call to a specified number when the phone goes off-hook, which is sometimes referred to as Hot Dialing.

Automatic Off-Hook Call Placement Parameters

As shown in the following list, you can specify an off-hook call contact, enable or disable the feature for each registration, and specify a protocol for the call.

Configuring these parameters will cause the phone to reboot.

call.autoOffHook.x.contact

Enter a SIP URL contact address. The contact must be an ASCII-encoded string containing digits, either the user part of a SIP URL (for example, 6416), or a full SIP URL (for example, 6416@polycom.com).

NULL (default)

call.autoOffHook.x.enabled

0 (default) - No call is placed automatically when the phone goes off hook, and the other parameters are ignored.

1 - When the phone goes off hook, a call is automatically placed to the contact you specify in call.autoOffHook.x.contact and using the protocol you specify in call.autoOffHook.x.protocol.

call.autoOffHook.x.protocol

Specify the calling protocol. If no protocol is specified, the phone uses the protocol specified by call.autoRouting.preferredProtocol. If a line is configured for a single protocol, the configured protocol is used.

NULL (default)

SIP

H323

Directed Call Pickup

Directed call pickup enables users to pick up incoming calls to another phone by dialing the extension of that phone.

This feature requires support from a SIP server and setup of this feature depends on the SIP server. For example, while some SIP servers implement directed call pick-up using a star-code sequence, others implement the feature using network signaling.
Directed Call Pickup Parameters

You can configure Directed Call Pickup using parameters in this section.

The parameters you use to configure this feature depends on your call server. To enable or disable this feature for Sylantro call servers, set the parameter `feature.directedCallPickup.enabled` to 1.

To configure this feature for all other call servers, use the following parameters:

- `call.directedCallPickupMethod`
- `call.directedCallPickupString`

Note that the pickup string can be different for different call servers, so check with your call server provider if you configure legacy mode for directed call pickup.

The following list includes the configuration parameters for the directed call pick-up feature.

**feature.directedCallPickup.enabled**

- 0 (default) - Disables the directed call pickup feature.
- 1 - Enables the directed call pickup feature.

Change causes system to restart or reboot.

**call.directedCallPickupMethod**

Specifies how the phone performs a directed call pick-up from a BLF contact.

- legacy (default) - Indicates that the phone uses the method specified in `call.directedCallPickupString`.
- native - Indicates that the phone uses a native protocol method (in this case SIP INVITE with the Replaces header).

**call.directedCallPickupString**

The star code to initiate a directed call pickup.

- *97 (default)

**Note:** The default value supports the BroadWorks calls server only. You must change the value if your organization uses a different call server.

**voIpProt.SIP.strictReplacesHeader**

This parameter applies only to directed call pick-up attempts initiated against monitored BLF resources.

- 1 (default) - The phone requires call-id, to-tag, and from-tag to perform a directed call-pickup when `call.directedCallPickupMethod` is configured as native.
- 0 - Call pick-up requires a call id only.
Group Call Pickup
This feature enables users to pick up incoming calls to any phone within a predefined group of phones, without dialing the extension of another phone.

Group Call Pickup Parameters
This feature requires support from a SIP server and setup of this feature depends on the SIP server.
For example, while some SIP servers implement group call pick-up using a particular star-code sequence, others implement the feature using network signaling.

feature.groupCallPickup.enabled
Enable or disable SIP-B Group Call Pickup.
0 (default) - Disabled
1 - Enabled
Change causes system to restart or reboot.

Multiple Line Registrations
Poly phones can have multiple line registrations.
Each registration requires an address or phone number.
When multiple registrations are available, users can select which registration to use for certain features, including which registration to use for outgoing calls or when initiating new instant messages.

Note: You must use a unique address or a phone number for each registration. Using the same address or phone number for multiple registrations might cause unexpected behavior.

Maximum Number of Registrations
The maximum number of registrations vary by phone and are listed in the following table.

Maximum Number of Registrations Per Phone

<table>
<thead>
<tr>
<th>Phone Model Name</th>
<th>Maximum Registrations</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCX 400</td>
<td>34</td>
</tr>
<tr>
<td>CCX 400</td>
<td>34</td>
</tr>
<tr>
<td>CCX 400</td>
<td>34</td>
</tr>
<tr>
<td>CCX 400</td>
<td>34</td>
</tr>
</tbody>
</table>
**Multiple Line Registrations Parameters**

Each registration can be mapped to one or more line keys, however, a line key can be used for only one registration.

The maximum number of call appearances you can set varies by phone model.

**reg.x.acd-agent-available**

- 0 (default) - The ACD feature is disabled for registration.
- 1 - If both ACD login/logout and agent available are set to 1 for registration x, the ACD feature is enabled for that registration.

**reg.x.acd-login-logout reg.x.acd-agent-available**

- 0 (default) - The ACD feature is disabled for registration.
- 1 - If both ACD login/logout and agent available are set to 1 for registration x, the ACD feature is enabled for that registration.

**reg.x.address**

The user part (for example, 1002) or the user and the host part (for example, 1002@polycom.com) of the registration SIP URI.

Null (default)

string address

**reg.x.advancedConference.maxParticipants**

Sets the maximum number of participants allowed in a push to conference for advanced conference calls. The number of participants configured must match the number of participants allowed on the ALU CTS.

- 3 (default)
- 0 - 25

**reg.x.advancedConference.pushToConference**

- 0 (default) - Disable push-to-conference functionality.
- 1 - Enable push-to-conference functionality.

**reg.x.advancedConference.subscribeForConfEvents**

- 1 (default) - Conference participants to receive notifications for conference events is enabled.
- 0 - Conference participants to receive notifications for conference events is disabled.
reg.x.advancedConference.subscribeForConfEventsOnCCPE
1 (default) - Enable the conference host to receive notifications for conference events.
0 - Disable the conference host to receive notifications for conference events.

reg.x.auth.domain
The domain of the authorization server that is used to check the user names and passwords.
Null (default) string

reg.x.auth.optimizedInFailover
The destination of the first new SIP request when failover occurs.
0 (default) - The SIP request is sent to the server with the highest priority in the server list.
1 - The SIP request is sent to the server which sent the proxy authentication request.

reg.x.auth.password
The password to be used for authentication challenges for this registration.
Null (default)
string - It overrides the password entered into the Authentication submenu on the Settings menu of the phone.

reg.x.auth.useLoginCredentials
0 - (default) The Login credentials are not used for authentication to the server on registration x.
1 - The login credentials are used for authentication to the server.

reg.x.auth.userId
User ID to be used for authentication challenges for this registration.
Null (default)
string - If the User ID is non-Null, it overrides the user parameter entered into the Authentication submenu on the Settings menu of the phone.

reg.x.bargeInEnabled
0 (default) - barge-in is disabled for line x.
1 - barge-in is enabled (remote users of shared call appearances can interrupt or barge in to active calls).

reg.x.bridgeInEnabled
0 (default) - Bridge In feature is disabled.
1 - Bridge In feature is enabled.

reg.x.broadsoft.userId
Enter the BroadSoft user ID to authenticate with the BroadSoft XSP service interface.
Null (default)
string

reg.x.broadsoft.useXspCredentials
If this parameter is disabled, the phones use standard SIP credentials to authenticate.
1 (default) - Use this value, if phone lines are registered with a server running BroadWorks R19 or earlier.
0 - Set to 0, if phone lines are registered with a server running BroadWorks R19 SP1 or later.

reg.x.broadsoft.xsp.password
Enter the password associated with the BroadSoft user account for the line. Required only when
reg.x.broadsoft.useXspCredentials=1.
Null (default)
string

reg.x.callsPerLineKey
Set the maximum number of concurrent calls for a single registration x. This parameter applies
to all line keys using registration x. If registration x is a shared line, an active call counts as a call
appearance on all phones sharing that registration.
This per-registration parameter overrides call.callsPerLineKey.
24 (default)
1 - 24

reg.x.displayName
The display name used in SIP signaling.
Null (default)
UTF-8 encoded string

reg.x.enablePvtHoldSoftKey
This parameter applies only to shared lines.
0 (default) - To disable user on a shared line to hold calls privately.
1 - To enable users on a shared line to hold calls privately.

reg.x.enablePvtHoldSoftKey
This parameter applies only to shared lines.
0 (default) - To disable user on a shared line to hold calls privately.
1 - To enable users on a shared line to hold calls privately.

reg.x.enhancedCallPark.enabled
0 (default) - To disable the BroadWorks Enhanced Call Park feature.
1 - To enable the BroadWorks Enhanced Call Park feature.

\texttt{reg.x.filterReflectedBlaDialogs}

1 (default) - bridged line appearance NOTIFY messages are ignored.
0 - bridged line appearance NOTIFY messages is not ignored

\texttt{reg.x.fwd.busy.contact}

The forward-to contact for calls forwarded due to busy status.
Null (default) - The contact specified by \texttt{divert.x.contact} is used.
string - The contact specified by \texttt{divert.x.contact} is not used

\texttt{reg.x.fwd.busy.status}

0 (default) - Incoming calls that receive a busy signal is not forwarded
1 - Busy calls are forwarded to the contact specified by \texttt{reg.x.fwd.busy.contact}.

\texttt{reg.x.fwd.noanswer.contact}

Null (default) - The forward-to contact specified by \texttt{divert.x.contact} is used.
string - The forward to contact used for calls forwarded due to no answer.

\texttt{reg.x.fwd.noanswer.contact}

Null (default) - The forward-to contact specified by \texttt{divert.x.contact} is used.
string - The forward to contact used for calls forwarded due to no answer.

\texttt{reg.x.fwd.noanswer.ringCount}

The number of seconds the phone should ring for before the call is forwarded because of no answer. The maximum value accepted by some call servers is 20.
0 - (default)
1 to 65535
**reg.x.fwd.noanswer.ringCount**

The number of seconds the phone should ring for before the call is forwarded because of no answer. The maximum value accepted by some call servers is 20.

0 - (default)
1 to 65535

**reg.x.fwd.noanswer.status**

0 (default) - The calls are not forwarded if there is no answer.
1 - The calls are forwarded to the contact specified by `reg.x.noanswer.contact` after ringing for the length of time specified by `reg.x.fwd.noanswer.ringCount`.

**reg.x.fwd.noanswer.status**

0 (default) - The calls are not forwarded if there is no answer.
1 - The calls are forwarded to the contact specified by `reg.x.noanswer.contact` after ringing for the length of time specified by `reg.x.fwd.noanswer.ringCount`.

**reg.x.gruu**

1 - The phone sends sip.instance in the REGISTER request.
0 (default) - The phone does not send sip.instance in the REGISTER request.

**reg.x.gruu**

Specify if the phone sends sip.instance in the REGISTER request.

0 (default) - Disabled
1 - Enabled

**reg.x.header.pearlymedia.support**

0 (Default) - The p-early-media header is not supported on the specified line registration.
1 - The p-early-media header is supported by the specified line registration.

**reg.X.insertOBPAddressInRoute**

1 (Default) - The outbound proxy address is added as the topmost route header.
0 - The outbound proxy address is not added to the route header.

**reg.x.label**

The text label that displays next to the line key for registration x.

The maximum number of characters for this parameter value is 256; however, the maximum number of characters that a phone can display on its user interface varies by phone model and by the width of the characters you use. Parameter values that exceed the phone's maximum display length are truncated by ellipses (...). The rules for parameter up.cfgLabelElide determine how the label is truncated.

Null (default) - the label is determined as follows:
If `reg.1.useteluriAsLineLabel=1`, then the tel URI/phone number/address displays as the label.

If `reg.1.useteluriAsLineLabel=0`, then the value for `reg.x.displayName`, if available, displays as the label. If `reg.x.displayName` is unavailable, the user part of `reg.x.address` is used.

UTF-8 encoded string

**reg.x.line.y.label**

Configure a unique line label for a shared line that has multiple line key appearances. This parameter takes effect when `up.cfgUniqueLineLabel=1`. If `reg.x.linekeys=1`, this parameter does not have any effect.

- `x` = the registration index number starting from 1.
- `y` = the line index from 1 to the value set by `reg.x.linekeys`. Specifying a string sets the label used for the line key registration on phones with multiple line keys.

If no parameter value is set for `reg.x.line.y.label`, the phone automatically numbers multiple lines by prepending "<y>" where `<y>` is the line index from 1 to the value set by `reg.x.linekeys`.

- The following examples show labels for line 1 on a phone with user registration 1234, where `reg.x.linekeys=2`:
  - If no label is configured for registration, the labels are "1_1234" and "2_1234".
  - If `reg.1.line.1.label=Polycom` and `reg.1.line.2.label=CCX`, the labels display as 'Polycom' and 'CCX'.

**reg.x.lineAddress**

The line extension for a shared line. This parameter applies to private lines and BroadSoft call park and retrieve. If there is no extension provided for this parameter, the call park notification is ignored for the shared line.

Null (default)
String

**reg.x.lineKeys**

Specify the number of line keys to use for a single registration. The maximum number of line keys you can use per registration depends on your phone model.
reg.x.lisd泄項

This parameter sets the value of the location policy disclaimer. For example, the disclaimer may be “Warning: If you do not provide a location, emergency services may be delayed in reaching your location should you need to call for help.”

Null (default)
string, 0 to 256 characters

reg.x.musicOnHold.uri

A URI that provides the media stream to play for the remote party on hold.

Null (default) - This parameter does not overrides voIpProt.SIP.musicOnHold.uri.
a SIP URI - This parameter overrides voIpProt.SIP.musicOnHold.uri.

reg.x.offerFullCodecListUponResume

1 (default) - The phone sends full audio and video capabilities after resuming a held call irrespective of the audio and video capabilities negotiated at the initial call answer.
0 - The phone does not send full audio and video capabilities after resuming a held call.

reg.x.offerFullCodecListUponHold

0 (default) - The phone doesn’t send full audio and video capabilities after a hold call.
1 - The phone sends full audio and video capabilities after a hold call.

reg.x.outboundProxy.address

The IP address or hostname of the SIP server to which the phone sends all requests.

Null (default)
IP address or hostname

reg.x.outboundProxy.failOver.failBack.mode

The mode for failover failback (overrides reg.x.server.y.failOver.failBack.mode).
duration - (default) The phone tries the primary server again after the time specified by reg.x.outboundProxy.failOver.failBack.timeout expires.
newRequests - All new requests are forwarded first to the primary server regardless of the last used server.
DNSTTL - The phone tries the primary server again after a timeout equal to the DNS TTL configured for the server that the phone is registered to.

reg.x.outboundProxy.failOver.failBack.timeout

3600 (default) - The time to wait (in seconds) before failback occurs (overrides reg.x.server.y.failOver.failBack.timeout).
0, 60 to 65535 - The phone does not fail back until a failover event occurs with the current server.

**reg.x.outboundProxy.failOver.failRegistrationOn**

1 (default) - The reRegisterOn parameter is enabled, the phone silently invalidates an existing registration.

0 - The reRegisterOn parameter is enabled, existing registrations remain active.

**reg.x.outboundProxy.failOver.onlySignalWithRegistered**

1 (default) - The reRegisterOn and failRegistrationOn parameters are enabled, no signaling is accepted from or sent to a server that has failed until failback is attempted or failover occurs.

0 - The reRegisterOn and failRegistrationOn parameters are enabled, signaling is accepted from and sent to a server that has failed.

**reg.x.outboundProxy.failOver.reRegisterOn**

This parameter overrides **reg.x.server.y.failOver.reRegisterOn**.

0 (default) - The phone won't attempt to register with the secondary server.

1 - The phone attempts to register with (or via, for the outbound proxy scenario), the secondary server.

**reg.x.outboundProxy.port**

The port of the SIP server to which the phone sends all requests.

0 - (default)

1 to 65535

**reg.x.outboundProxy.transport**

The transport method the phone uses to communicate with the SIP server.

- DNSnaptr (default)
- DNSnaptr, TCPpreferred, UDPOnly, TLS, TCPOnly

**reg.x.path**

0 (Default) - The path extension header field in the Register request message is not supported for the specific line registration.

1 - The phone supports and provides the path extension header field in the Register request message for the specific line registration.

**reg.x.protocol.SIP**

1 (default) - SIP signaling is enabled for this registration.

0 - SIP signaling is not enabled for this registration.
**reg.x.proxyRequire**

Null (default) - No Proxy-Require is sent.
string - Needs to be entered in the Proxy-Require header.

**reg.x.regevent**

0 (default) - The phone is not subscribed to registration state change notifications for the specific phone line.
1 - The phone is subscribed to registration state change notifications for the specific phone line.
This parameter overrides the global parameter voIpProt.SIP.regevent.

**reg.x.rejectNDUBInvite**

Specify whether or not the phone accepts a call for a particular registration in case of a Network Determined User Busy (NDUB) event advertised by the SIP server.
0 (Default) - If an NDUB event occurs, the phone does not reject the call.
1 - If an NDUB event occurs, the phone rejects the call with a 603 Decline response code.

**reg.x.ringType**

The ringer to be used for calls received by this registration. The default is the first non-silent ringer.
If you use the configuration parameters ringer13 and ringer14 on a single registered line, the phone plays SystemRing.wav.
default (default)
ring1 to ringer24

**reg.x.ringType**

The ringer to be used for calls received by this registration.
ring2 (default) - Is the first non-silent ringer.
ring1 to ringer24 - To play ringer on a single registered line.

**reg.x.server.y.address**

If this parameter is set, it takes precedence even if the DHCP server is available.
Null (default) - SIP server does not accepts registrations.
IP address or hostname - SIP server that accepts registrations. If not Null, all of the parameters in this list override the parameters specified in voIpProt.server.*

**reg.x.server.y.expires**

The phone's requested registration period in seconds.
The period negotiated with the server may be different. The phone attempts to re-register at the beginning of the overlap period.
3600 - (default)
positive integer, minimum 10

**reg.x.server.y.expires.lineSeize**

Requested line-seize subscription period.

30 - (default)
0 to 65535

**reg.x.server.y.expires.overlap**

The number of seconds before the expiration time returned by server x at which the phone should try to re-register.

The phone tries to re-register at half the expiration time returned by the server if the server value is less than the configured overlap value.

60 (default)
5 to 65535

**reg.x.server.y.failOver.failBack.mode**

duration (default) - The phone tries the primary server again after the time specified by `reg.x.server.y.failOver.failBack.timeout`.

ewRequests - All new requests are forwarded first to the primary server regardless of the last used server.

DNSTTL - The phone tries the primary server again after a timeout equal to the DNS TTL configured for the server that the phone is registered to.

registration - The phone tries the primary server again when the registration renewal signaling begins.

This parameter overrides `voIPProt.server.x.failOver.failBack.mode`

**reg.x.server.y.failOver.failBack.timeout**

3600 (default) - The time to wait (in seconds) before failback occurs.
0 - The phone does not fail back until a failover event occurs with the current server.
60 to 65535 - If set to Duration, the phone waits this long after connecting to the current working server before selecting the primary server again.

**reg.x.server.y.failOver.failRegistrationOn**

1 (default) - The reRegisterOn parameter is enabled, the phone silently invalidates an existing registration (if it exists) at the point of failing over.
0 - The reRegisterOn parameter is disabled, existing registrations remain active.

**reg.x.server.y.failOver.onlySignalWithRegistered**

1 (default) - Set to this value and reRegisterOn and failRegistrationOn parameters are enabled, no signaling is accepted from or sent to a server that has failed until failback is attempted or failover occurs. If the phone attempts to send signaling associated with an existing call via an
unregistered server (for example, to resume or hold a call), the call ends. No SIP messages are sent to the unregistered server.

0 - Set to this value and reRegisterOn and failRegistrationOn parameters are enabled, signaling is accepted from and sent to a server that has failed (even though failback hasn't been attempted or failover hasn't occurred).

\textbf{reg.x.server.y.failOver.reRegisterOn}

0 (default) - The phone does not attempt to register with the secondary server, since the phone assumes that the primary and secondary servers share registration information.

1 - The phone attempts to register with (or via, for the outbound proxy scenario), the secondary server. If the registration succeeds (a 200 OK response with valid expires), signaling proceeds with the secondary server.

This parameter overrides voIpProt.server.x.failOver.reRegisterOn.

\textbf{reg.x.server.y.port}

Null (default) - The port of the SIP server does not specifies registrations.

0 - The port used depends on reg.x.server.y.transport.

1 to 65535 - The port of the SIP server that specifies registrations.

\textbf{reg.x.server.y.register}

1 (default) - Calls can not be routed to an outbound proxy without registration.

0 - Calls can be routed to an outbound proxy without registration.

See voIpProt.server.x.register for more information, see SIP Server Fallback Enhancements on Polycom Phones - Technical Bulletin 5844 on Polycom Engineering Advisories and Technical Notifications.

\textbf{reg.x.server.y.registerRetry.baseTimeOut}

For registered line x, set y to the maximum time period the phone waits before trying to re-register with the server. Used in conjunction with reg.x.server.y.registerRetry.maxTimeOut to determine how long to wait.

60 (default)

10 - 120 seconds

\textbf{reg.x.server.y.registerRetry.maxTimeout}

For registered line x, set y to the maximum time period the phone waits before trying to re-register with the server. Use in conjunction with reg.x.server.y.registerRetry.baseTimeOut to determine how long to wait. The algorithm is defined in RFC 5626.

180 - (default)

60 - 1800 seconds

\textbf{reg.x.server.y.retryMaxCount}

The number of retries attempted before moving to the next available server.
3 - (default)
0 to 20 - 3 is used when the value is set to 0.

**reg.x.server.y.retryTimeOut**

0 (default) - Use standard RFC 3261 signaling retry behavior.
0 to 65535 - The amount of time (in milliseconds) to wait between retries.

**reg.x.server.y.specialInterop**

Specify the server-specific feature set for the line registration.
Standard (Default)
All other phones:
Standard
GENBAND
ALU-CTS
ocs2007r2
lcs2005

**reg.x.server.y.subscribe.expires**

The phone’s requested subscription period in seconds after which the phone attempts to resubscribe at the beginning of the overlap period.
3600 seconds - (default)
10 - 2147483647 (seconds)
You can use this parameter in conjunction with reg.x.server.y.subscribe.expires.overlap.

**reg.x.server.y.subscribe.expires**

The phone’s requested subscription period in seconds after which the phone attempts to resubscribe at the beginning of the overlap period.
3600 seconds - (default)
10 - 2147483647 (seconds)
You can use this parameter in conjunction with reg.x.server.y.subscribe.expires.overlap.

**reg.x.server.y.subscribe.expires.overlap**

The number of seconds before the expiration time returned by server x after which the phone attempts to resubscribe. If the server value is less than the configured overlap value, the phone tries to resubscribe at half the expiration time returned by the server.
60 seconds (default)
5 - 65535 seconds
**reg.x.server.y.subscribe.expires.overlap**

The number of seconds before the expiration time returned by server x after which the phone attempts to resubscribe. If the server value is less than the configured overlap value, the phone tries to resubscribe at half the expiration time returned by the server.

60 seconds (default)
5 - 65535 seconds

**reg.x.server.y.transport**

The transport method the phone uses to communicate with the SIP server.

DNSnaptr (default) - If reg.x.server.y.address is a hostname and reg.x.server.y.port is 0 or Null, do NAPTR then SRV look-ups to try to discover the transport, ports and servers, as per RFC 3263. If reg.x.server.y.address is an IP address, or a port is given, then UDP is used.

TCPpreferred - TCP is the preferred transport; UDP is used if TCP fails.

UDPOnly - Only UDP is used.

TLS - If TLS fails, transport fails. Leave port field empty (defaults to 5061 ) or set to 5061 .

TCPOnly - Only TCP is used.

**reg.x.server.y.useOutboundProxy**

1 (default) - Enables to use the outbound proxy specified in reg.x.outboundProxy.address for server x.

0 - Disable to use the outbound proxy specified in reg.x.outboundProxy.address for server x.

**reg.x.serverFeatureControl.callRecording**

1 (default) - BroadSoft BroadWorks v20 call recording feature for individual phone lines is enabled.

0 - BroadSoft BroadWorks v20 call recording feature for individual phone lines is disabled.

**reg.x.serverFeatureControl.callRecording**

1 (default) - BroadSoft BroadWorks v20 call recording feature for individual phone lines is enabled.

0 - BroadSoft BroadWorks v20 call recording feature for individual phone lines is disabled.

**reg.x.serverFeatureControl.cf**

This parameter overrides voIpProt.SIP.serverFeatureControl.cf .

0 (default) - The server-based call forwarding is disabled.

1 - server based call forwarding is enabled.

Change causes system to restart or reboot.
**reg.x.serverFeatureControl.cf**

This parameter overrides `voIpProt.SIP.serverFeatureControl.cf`.

0 (default) - The server-based call forwarding is disabled.
1 - server based call forwarding is enabled.
Change causes system to restart or reboot.

**reg.x.serverFeatureControl.dnd**

This parameter overrides `voIpProt.SIP.serverFeatureControl.dnd`.

0 (default) - server-based do-not-disturb (DND) is disabled.
1 - server-based DND is enabled and the call server has control of DND.
Change causes system to restart or reboot.

**reg.x.serverFeatureControl.localProcessing.cf**

This parameter overrides `voIpProt.SIP.serverFeatureControl.localProcessing.cf`.

0 (default) - If `reg.x.serverFeatureControl.cf` is set to 1 the phone does not perform local Call Forward behavior.
1 - The phone performs local Call Forward behavior on all calls received.

**reg.x.serverFeatureControl.localProcessing.dnd**

This parameter overrides `voIpProt.SIP.serverFeatureControl.localProcessing.dnd`.

0 (default) - If `reg.x.serverFeatureControl.dnd` is set to 1, the phone does not perform local DND call behavior.
1 - The phone performs local DND call behavior on all calls received.

**reg.x.serverFeatureControl.securityClassification**

0 (default) - The visual security classification feature for a specific phone line is disabled.
1 - The visual security classification feature for a specific phone line is enabled.
**reg.x.serverFeatureControl.signalingMethod**

Controls the method used to perform call forwarding requests to the server.

- serviceMsForwardContact (default)
- string

**reg.x.srtp.enable**

- 1 (default) - The registration accepts SRTP offers.
- 0 - The registration always declines SRTP offers.

Change causes system to restart or reboot.

**reg.x.srtp.offer**

This parameter applies to the registration initiating (offering) a phone call.

- 0 (default) - No secure media stream is included in SDP of a SIP INVITE.
- 1 - The registration includes a secure media stream description along with the usual non-secure media description in the SDP of a SIP INVITE.

Change causes system to restart or reboot.

**reg.x.srtp.require**

- 0 (default) - Secure media streams are not required.
- 1 - The registration is only allowed to use secure media streams.

Change causes system to restart or reboot.

**reg.x.srtp.simplifiedBestEffort**

This parameter overrides `sec.srtp.simplifiedBestEffort`.

- 0 (default) - SRTP negotiation compliant with Microsoft Session Description Protocol Version 2.0 Extensions is not supported.
- 1 - SRTP negotiation compliant with Microsoft Session Description Protocol Version 2.0 Extensions is supported.

**reg.x.strictLineSeize**

- 0 (default) - Dial prompt is provided immediately without waiting for a successful OK from the call server.
- 1 - The phone is forced to wait for 200 OK on registration x when receiving a TRYING notify.

This parameter overrides `volpProt.SIP.strictLineSeize` for registration x.

**reg.x.tcpFastFailover**

- 0 (default) - A full 32 second RFC compliant timeout is used.
- 1 - Failover occurs based on the values of `reg.x.server.y.retryMaxCount` and `voIpProt.server.x.retryTimeOut`.
**reg.x.thirdPartyName**

Null (default) - In all other cases.

string address - This field must match the `reg.x.address` value of the registration which makes up the part of a bridged line appearance (BLA).

**reg.x.type**

private (default) - Use standard call signaling.

shared - Use augment call signaling with call state subscriptions and notifications and use access control for outgoing calls.

**reg.x.useCompleteUriForRetrieve**

This parameters overrides `voipPort.SIP.useCompleteUriForRetrieve`.

1 (default) - The target URI in BLF signaling uses the complete address as provided in the XML dialog document.

0 - Only the user portion of the XML dialog document is used and the current registrar's domain is appended to create the full target URI.

**voipProt.server.x.address**

The IP address or hostname and port of a SIP server that accepts registrations. Multiple servers can be listed starting with x=1 to 4 for fault tolerance.

Null (default), IP address, or hostname

**voIpProt.server.x.expires**

The phone's requested registration period in seconds.

3600 (default)

positive integer, minimum 10

The period negotiated with the server may be different. The phone attempts to re-register at the beginning of the overlap period. For example, if `expires="300"` and `overlap="5"`, the phone re-registers after 295 seconds (300-5).
voIpProt.server.x.expires
The phone's requested registration period in seconds. Note: The period negotiated with the server may be different. The phone will attempt to re-register at the beginning of the overlap period.
3600 (default)
positive integer, minimum 10

voIpProt.server.x.expires.lineSeize
Requested line-seize subscription period.
30 (default)
positive integer, minimum 10

voIpProt.server.x.expires.lineSeize
Requested line-seize subscription period.
30 (default)
positive integer, minimum 0 was 10

voIpProt.server.x.expires.overlap
The number of seconds before the expiration time returned by server x at which the phone should try to re-register. If the server value is less than the configured overlap value, the phone tries to re-register at half the expiration time returned by the server.
60 (default)
5 to 65536

voIpProt.server.x.expires.overlap
The number of seconds before the expiration time returned by server x at which the phone should try to re-register. The phone will try to re-register at half the expiration time returned by the server if the server value is less than the configured overlap value.
60 (default)
5 to 65535

voIpProt.server.x.failOver.failBack.mode
Specify the failover failback mode.
duration (default) - The phone tries the primary server again after the time specified by voIpProt.server.x.failOver.failBack.timeout
newRequests - All new requests are forwarded first to the primary server regardless of the last used server.
DNSTTL - The phone tries the primary server again after a timeout equal to the DNS TTL configured for the server that the phone is registered to.
registration - The phone tries the primary server again when the registration renewal signaling begins.
**voIpProt.server.x.failOver.failBack.timeout**

If `voIpProt.server.x.failOver.failBack.mode` is set to duration, this is the time in seconds after failing over to the current working server before the primary server is again selected as the first server to forward new requests to. Values between 1 and 59 result in a timeout of 60 and 0 means do not fail-back until a fail-over event occurs with the current server.

3600 (default)

0, 60 to 65535

**voIpProt.server.x.failOver.failRegistrationOn**

1 (default) - When set to 1, and the reRegisterOn parameter is enabled, the phone silently invalidates an existing registration (if it exists), at the point of failing over.

0 - When set to 0, and the reRegisterOn parameter is enabled, existing registrations remain active. This means that the phone attempts failback without first attempting to register with the primary server to determine if it has recovered.

**voIpProt.server.x.failOver.onlySignalWithRegistered**

1 (default) - When set to 1, and the reRegisterOn and failRegistrationOn parameters are enabled, no signaling is accepted from or sent to a server that has failed until failback is attempted or failover occurs. If the phone attempts to send signaling associated with an existing call via an unregistered server (for example, to resume or hold a call), the call ends. No SIP messages are sent to the unregistered server.

0 - When set to 0, and the reRegisterOn and failRegistrationOn parameters are enabled, signaling is accepted from and sent to a server that has failed (even though failback hasn't been attempted or failover hasn't occurred).

**voIpProt.server.x.failOver.reRegisterOn**

0 (default) - When set to 0, the phone won't attempt to register with the second.

1 - When set to 1, the phone attempts to register with (or by, for the outbound proxy scenario), the secondary server. If the registration succeeds (a 200 OK response with valid expires), signaling proceeds with the secondary server.

**voIpProt.server.x.port**

The port of the server that specifies registrations.

0 (default) - If 0, the port used depends on `voIpProt.server.x.transport`.

1 to 65535

**voIpProt.server.x.protocol.SIP**

1 (default) - Server is a SIP proxy/registrar

0 - If set to 0, and the server is confirmed to be a SIP server, then the value is assumed to be 1.

**voIpProt.server.x.register**

1 (default) - Calls can not be routed to an outbound proxy without registration.

0 - Calls can be routed to an outbound proxy without registration.
For more information, see Technical Bulletin 5844: SIP Server Fallback Enhancements on Polycom Phones.

voIpProt.server.x.registerRetry.baseTimeOut
The base time period to wait before a registration retry. Used in conjunction with voIpProt.server.x.registerRetry.maxTimeOut to determine how long to wait. The algorithm is defined in RFC 5626.
If both parameters voIpProt.server.x.registerRetry.baseTimeOut and reg.x.server.y.registerRetry.baseTimeOut are set, the value of reg.x.server.y.registerRetry.baseTimeOut takes precedence.

60 - (default)
10 - 120

voIpProt.server.x.registerRetry.maxTimeOut
The maximum time period to wait before a registration retry. Used in conjunction with voIpProt.server.x.registerRetry.maxTimeOut to determine how long to wait. The algorithm is defined in RFC 5626.
If both parameters voIpProt.server.x.registerRetry.maxTimeOut and reg.x.server.y.registerRetry.maxTimeOut are set, the value of reg.x.server.y.registerRetry.maxTimeOut takes precedence.

60 - (default)
10 - 1800

voIpProt.server.x.retryMaxCount
The number of retries that will be attempted before moving to the next available server.
3 (default)
0 to 20 - If set to 0, 3 is used.

voIpProt.server.x.retryTimeOut
0 (default) - Use standard RFC 3261 signaling retry behavior.
0 to 65535 - The amount of time (in milliseconds) to wait between retries.

voIpProt.server.x.specialInterop
Enables server-specific features for all registrations.
Standard (default)
All other phones =
Standard
GENBAND
GENBAND-A2
ALU-CTS
DT
voIpProt.server.x.subscribe.expires

The phone's requested subscription period in seconds after which the phone attempts to resubscribe at the beginning of the overlap period.
3600 - (default)
10 - 2147483647

voIpProt.server.x.subscribe.expires.overlap

The number of seconds before the expiration time returned by server x after which the phone attempts to resubscribe. If the server value is less than the configured overlap value, the phone tries to resubscribe at half the expiration time returned by the server.
60 - (default)
5 - 65535 seconds

voIpProt.server.x.transport

The transport method the phone uses to communicate with the SIP server.
Null or DNSnaptr (default) - If voIpProt.server.x.address is a hostname and voIpProt.server.x.port is 0 or Null, do NAPTR then SRV look-ups to try to discover the transport, ports and servers, as per RFC 3263. If voIpProt.server.x.address is an IP address, or a port is given, then UDP is used.
TCPpreferred - TCP is the preferred transport; UDP is used if TCP fails.
UDPOnly - Only UDP will be used.
TLS - If TLS fails, transport fails. Leave port field empty (will default to 5061) or set to 5061.
TCPOnly - Only TCP will be used.

voIpProt.server.x.useOutboundProxy

1 (default) - Enables to use the outbound proxy specified in voIpProt.SIP.outboundProxy.address for server x.
0 - Enables not to use the outbound proxy specified in voIpProt.SIP.outboundProxy.address for server x.

voIpProt.SIP.acd.signalingMethod

0 (default) - The 'SIP-B' signaling is supported. (This is the older ACD functionality.)
1 - The feature synchronization signaling is supported. (This is the new ACD functionality.)
Change causes system to restart or reboot.

voIpProt.SIP.acd.signalingMethod

0 (default) - The 'SIP-B' signaling is supported. (This is the older ACD functionality.)
1 - The feature synchronization signaling is supported. (This is the new ACD functionality.)
Change causes system to restart or reboot.

**voIpProt.SIP.alertInfo.x.class**

Alert-Info fields from INVITE requests are compared as many of these parameters as are specified (x=1, 2, ..., N) and if a match is found, the behavior described in the corresponding ring class is applied.

default (default)

See the list of ring classes in Ringtone Parameters.

**voIpProt.SIP.alertInfo.x.class**

Alert-Info fields from INVITE requests are compared as many of these parameters as are specified (x=1, 2, ..., N) and if a match is found, the behavior described in the corresponding ring class is applied.

default (default)

See the list of ring classes in Ringtone Parameters.

**voIpProt.SIP.alertInfo.x.value**

Specify a ringtone for single registered line using a string to match the Alert-Info header in the incoming INVITE.

NULL (default)

**voIpProt.SIP.alertInfo.x.value**

Specify a ringtone for single registered line using a string to match the Alert-Info header in the incoming INVITE.

NULL (default)

**voIpProt.SIP.alertInfo.x.value**

Specify a ringtone for single registered line using a string to match the Alert-Info header in the incoming INVITE.

NULL (default)

**voIpProt.SIP.allowTransferOnProceeding**

1 (default) - Transfer during the proceeding state of a consultation call is enabled.

0 - Transfer during the proceeding state of a consultation call is enabled
2 - Phones will accept an INVITE with replaces for a dialog in early state. This is needed when using transfer on proceeding with a proxying call server such as openSIPS, reSIProcate or SipXecs.

**voipProt.SIP.anat.enabled**

Enables or disables Alternative Network Address Types (ANAT).

0 (default) - ANAT is disabled.

1 - ANAT is enabled.

**voIpProt.SIP.authOptimizedInFailover**

0 (default) - The first new SIP request is sent to the server with the highest priority in the server list when failover occurs.

1 - The first new SIP request is sent to the server that sent the proxy authentication request when failover occurs.

**voIpProt.SIP.callee.SourcePreference**

Set priority order to display the callee's identity for outgoing calls.

Null (default)

Supported Headers Default Order: P-Asserted-Identity,Remote-Party-ID,From String

**voIpProt.SIP.Caller.SourcePreference**

Set priority order to display the caller's identity for incoming calls.

Null (default)

Supported Headers Default Order: P-Asserted-Identity,Remote-Party-ID,From String

**voIpProt.SIP.callinfo.precedence.overAlertinfo**

0 (default) - Give priority to call-info header with answer-after string over alert-info feature is disabled.

1 - Give priority to call-info header with answer-after string over alert-info feature is enabled.

**voIpProt.SIP.callinfo.precedence.overAlertinfo**

0 (default) - The alert-info is given priority over call-info header.

1 - The call-info header with answer-after string is given priority over alert-info header.

**voIpProt.SIP.CID.request.sourceSipMessage**

Specify which header in the SIP request to retrieve remote party caller ID from. You can use:

- voIpProt.SIP.callee.sourcePreference
- voIpProt.SIP.caller.sourcePreference
CALL CONTROLS

- voIpProt.SIP.CID.sourcePreference

UPDATE takes precedence over the value of this parameter.

NULL (default) - Remote party caller ID information from INVITE is used.

INVITE

PRACK

ACK

This parameter does not apply to shared lines.

voIpProt.SIP.CID.response.sourceSipMessage

Specify which header in the SIP request to retrieve remote party caller ID from. You can use:

- voIpProt.SIP.callee.sourcePreference
- voIpProt.SIP.caller.sourcePreference
- voIpProt.SIP.CID.sourcePreference

NULL (default) - The remote party caller ID information from the last SIP response is used.

100, 180, 183, 200

This parameter does not apply to shared lines.

voIpProt.SIP.CID.sourcePreference

Specify the priority order for the sources of caller ID information. The headers can be in any order.

Null (default) - Caller ID information comes from P-Asserted-Identity, Remote-Party-ID, and From in that order.

From, P-Asserted-Identity, Remote-Party-ID

P-Asserted-Identity, From, Remote-Party-ID

Supported Headers Default Order: P-Asserted-Identity, Remote-Party-ID, From

If voIpProt.SIP.Caller.SourcePreference or voIpProt.SIP.Callee.SourcePreference are configured then the order set by voIpProt.SIP.CID.sourcePreference is ignored.

voIpProt.SIP.compliance.RFC3261.validate.contentLanguage

1 (default) - Validation of the SIP header content language is enabled.

0 - Validation of the SIP header content language is disabled

voIpProt.SIP.compliance.RFC3261.validate.contentLength

1 (default) - Validation of the SIP header content length is enabled.

0 - Validation of the SIP header content length is disabled

voIpProt.SIP.compliance.RFC3261.validate.uriScheme

1 (default) - Validation of the SIP header URI scheme is enabled.
0 - Validation of the SIP header URI scheme is disabled

**voIpProt.SIP.conference.address**

Null (default) - Conferences are set up on the phone locally.

String 128 max characters - Enter a conference address. Conferences are set up by the server using the conferencing agent specified by this address. Acceptable values depend on the conferencing server implementation policy.

**voIpProt.SIP.conference.parallelRefer**

0 (default) - A parallel REFER is not sent to the call server.
1 - A parallel REFER is not sent to the call server.

Note: This parameter must be set for Siemens OpenScape Centralized Conferencing.

**voIpProt.SIP.connectionReuse.useAlias**

0 (default) - The alias parameter is not added to the via header
1 - The phone uses the connection reuse draft which introduces "alias".

**voIpProt.SIP.dialog.strictXLineID**

0 (default) - The phone will not look for x-line-id (call appearance index) in a SIP INVITE message.
1 - The phone will look for x-line-id (call appearance index) in a SIP INVITE message.

**voIpProt.SIP.dialog.usePvalue**

0 (default) - Phone uses a `pval` field name in Dialog.
1 - Phone uses a `pvalue` field name in Dialog.

**voIpProt.SIP.dialog.useSDP**

0 (default) - A new dialog event package draft is used (no SDP in dialog body).
1 - Use this setting to send SDP in the dialog body for backwards compatibility.

**voIpProt.SIP.dtmfViaSignaling.rfc2976**

Enable or disable DTMF relays for active SIP calls.
0 (default) - DTMF digit information is not sent
1 - DTMF digit information is sent in RFC2976 SIP INFO packets during a call.

Change causes system to restart or reboot.

**voIpProt.SIP.dtmfViaSignaling.rfc2976.nonLegacyEncoding**

Controls the behavior of the Star and Pound keys used for DTMF relays for active SIP calls.
0 (default) - The phone sends 10 when the Star key (*) is pressed and 11 when the Pound key (#) is pressed.
1 - The phone sends an asterisk (*) when the Star key is pressed and a hashtag (#) when the Pound key is pressed.
Change causes system to restart or reboot.

**voIpProt.SIP.enable**
A flag to determine if the SIP protocol is used for call routing, dial plan, DTMF, and URL dialing.
1 (default) - The SIP protocol is used.
0 - The SIP protocol is not used.
Change causes system to restart or reboot.

**voIpProt.SIP.failoverOn503Response**
A flag to determine whether or not to trigger a failover if the phone receives a 503 response. You must use a registration expiry of 66 seconds or greater for failover with a 503 response to work properly. This rule applies both to the phone configuration (`reg.x.server.y.expires` and `voIpProt.server.x.expires`) as well as the 200 OK register response from the server.
1 (default) - Enabled
0 - Disabled

**voIpProt.SIP.header.diversion.enable**
0 (default) - If set to 0, the diversion header is not displayed.
1 - If set to 1, the diversion header is displayed if received.
Change causes system to restart or reboot.

**voIpProt.SIP.header.diversion.list.useFirst**
1 (default) - If set to 1, the first diversion header is displayed.
0 - If set to 0, the last diversion header is displayed.
Change causes system to restart or reboot.

**voIpProt.SIP.header.pEarlyMedia.support**
0 (default) - The p-early-media header is not supported by the caller phone.
1 - The p-early-media header is supported by the caller phone.

**voIpProt.SIP.header.warning.codes.accept**
Specify a list of accepted warning codes.
Null (default) - All codes are accepted only codes between 300 and 399 are supported.
comma separated list

**voIpProt.SIP.header.warning.codes.accept**
Specify a list of accepted warning codes.
Null (default) - All codes are accepted. Only codes between 300 and 399 are supported.
For example, if you want to accept only codes 325 to 330:
```plaintext
voIpProt.SIP.header.warning.codes.accept=325,326,327,328,329,330
```

**voIpProt.SIP.header.warning.enable**
0 (default) - The warning header is not displayed.
1 - The warning header is displayed if received.

**voIpProt.SIP.ignore.pEarlyMediaInactive**
0 (default) – The phone does not ignore SIP messages received with “inactive” in the p-Early-Media header.
1 – The phone ignores SIP messages received with “inactive” in the p-Early-Media header on a non-active early dialog in case of forking and does not switch to a local ringback tone.
This parameter applies only when `voIpProt.SIP.header.pEarlyMedia.support` is enabled.

**voIpProt.SIP.IM.autoAnswerDelay**
The time interval from receipt of the instant message invitation to automatically accepting the invitation.
10 (default)
0 to 40

**voIpProt.SIP.IMS.enable**
This parameter applies to all registered or unregistered SIP lines on the phone.
0 (default) - The phone does not support IMS features introduced in UC Software 5.5.0.
1 - The phone supports IMS features introduced in UC Software 5.5.0.

**voIpProt.SIP.intercom.alertInfo**
The string you want to use in the Alert-Info header. You can use the following characters: '@', '-', '_' , '.' .
If you use any other characters, NULL, or empty spaces, the call is sent as normal without the Alert-Info header.
Intercom (default)
Alpha - Numeric string

**voIpProt.SIP.keepalive.sessionTimers**
0 (default) - The session timer is disabled.
1 - The session timer is enabled.

**voIpProt.SIP.lineSeize.retries**
Controls the number of times the phone will retry a notify when attempting to seize a line (BLA).
10 (default)
voIpProt.SIP.local.port
The local port for sending and receiving SIP signaling packets.
5060 - The value is used for the local port but is not advertised in the SIP signaling.
0 to 65535 - If set to 0, the 5060 value is used for the local port but is not advertised in the SIP signaling. For other values, that value is used for the local port and it is advertised in the SIP signaling.
Change causes system to restart or reboot.

voIpProt.SIP.looseContact
0 (default) - The port parameter is added to the contact header in TLS case.
1 - The port parameter is not added to the contact header or SIP messages.

voIpProt.SIP.noContactHeaderIn200OKForNotify
0 (default) – Disabled
Phone sends contact header in 200 ok for NOTIFY.
1 – Enabled
Phone doesn't send contact header in 200 ok for NOTIFY.

voIpProt.SIP.ms-forking
This parameter is applies when installing Microsoft Live Communications Server.
0 (default) - Support for MS-forking is disabled.
1 - Support for MS-forking is enabled.
Note: If any endpoint registered to the same account has MS-forking disabled, all other endpoints default back to non-forking mode. Windows Messenger does not use MS-forking so be aware of this behavior if one of the endpoints is using Windows Messenger.

voIpProt.SIP.musicOnHold.uri
A URI that provides the media stream to play for the remote party on hold. This parameter is used if reg.x.musicOnHold.uri is Null.
Null (default)
SIP URI

voIpProt.SIP.newCallOnUnRegister
1 (default) - The phone generate new Call-ID and From tag during re-registration.
0 - The phone does not generate new Call-ID and From tag during re-registration.

voIpProt.SIP.outboundProxy.address
The IP address or hostname of the SIP server to which the phone sends all requests.
Null (default)
IP address or hostname

**voIpProt.SIP.outboundProxy.failOver.failBack.mode**

Duration (default) - The phone tries the primary server again after the time specified by `reg.x.outboundProxy.failOver.failBack.timeout` expires.

newRequests - All new requests are forwarded first to the primary server regardless of the last used server.

DNSTTL - The phone tries the primary server again after a timeout equal to the DNS TTL configured for the server that the phone is registered to.

registration - The phone tries the primary server again when the registration renewal signaling begins.

**voIpProt.SIP.outboundProxy.failOver.failBack.timeout**

The time to wait (in seconds) before failback occurs (overrides `voIpProt.server.x.failOver.failBack.timeout`).

3600 (default) - If the fail back mode is set to Duration, the phone waits this long after connecting to the current working server before selecting the primary server again.

0, 60 to 65535 - If set to 0, the phone will not fail-back until a fail-over event occurs with the current server.

**voIpProt.SIP.outboundProxy.failOver.failRegistrationOn**

1 (default) - When set to 1, and the `reRegisterOn` parameter is enabled, the phone will silently invalidate an existing registration (if it exists), at the point of failing over.

0 - When set to 0, and the `reRegisterOn` parameter is enabled, existing registrations will remain active. This means that the phone will attempt failback without first attempting to register with the primary server to determine if it has recovered.

**Note:** `voIpProt.SIP.outboundProxy.failOver.reRegisterOn` must be enabled.

**voIpProt.SIP.outboundProxy.failOver.onlySignalWithRegistered**

1 (default) - No signaling is accepted from or sent to a server that has failed until failback is attempted or failover occurs.

0 - Signaling will be accepted from and sent to a server that has failed (even though failback hasn't been attempted or failover hasn't occurred). This parameter overrides `voIpProt.server.x.failOver.onlySignalWithRegistered`.

**Note:** `reRegisterOn` and `failRegistrationOn` parameters must be enabled.

**voIpProt.SIP.outboundProxy.failOver.reRegisterOn**

This parameter overrides the `voIpProt.server.x.failOver.reRegisterOn`.

0 (default) - The phone won't attempt to register with the secondary server, since the phone will assume that the primary and secondary servers share registration information.
1 - The phone will attempt to register with the secondary server. If the registration succeeds signaling will proceed with the secondary server.

**voIpProt.SIP.outboundProxy.port**

The port of the SIP server to which the phone sends all requests.

0 (default)

0 to 65535

**voIpProt.SIP.outboundProxy.transport**

DNSnaptr (default) - If `reg.x.outboundProxy.address` is a hostname and `reg.x.outboundProxy.port` is 0 or Null, do NAPTR then SRV look-ups to try to discover the transport, ports and servers, as per RFC 3263. If `reg.x.outboundProxy.address` is an IP address, or a port is given, then UDP is used.

TCPpreferred - TCP is the preferred transport, UDP is used if TCP fails.

UDPOnly - Only UDP will be used.

TLS - If TLS fails, transport fails. Leave port field empty (will default to 5061) or set to 5061.

TCPOnly - Only TCP will be used.

**voIpProt.SIP.pingInterval**

The number in seconds to send PING message.

0 (default) - This feature is disabled.

0 to 3600 - This feature is enabled.

**voIpProt.SIP.pingMethod**

The ping method to be used.

PING (default)

OPTIONS

**voIpProt.SIP.presence.nortelShortMode**

This parameter is required when using the Presence feature with an Avaya or Ribbon Communications server.

0 (default)

1 - Different headers are sent in SUBSCRIBE when used feature with an Avaya or Ribbon Communications server. Support is indicated by adding a header `Accept-Encoding: x-nortel-short`. A PUBLISH is sent to indicate the status of the phone.

Change causes system to restart or reboot.

**voIpProt.SIP.regevent**

0 (default) - The phone is not subscribed to registration state change notifications for all phone lines.

1 - The phone is subscribed to registration state change notifications for all phone lines.
This parameter is overridden by the per-phone parameter reg.x.regevent.

**voIpProt.SIP.rejectNDUBInvite**

Specify whether or not the phone accepts a call for all registrations in case of a Network Determined User Busy (NDUB) event advertised by the SIP server.

0 (default) - If an NDUB event occurs, the phone does not reject the call for all line registrations.

1 - If an NDUB event occurs, the phone rejects the call with a 603 Decline response code for all line registrations.

**voIpProt.SIP.renewSubscribeOnTLSRefresh**

1 (default) - For an as-feature-event, the SUBSCRIBE message is sent along with the RE-REGISTER when Transport Layer Security (TLS) breaks.

0 - The SUBSCRIBE and RE-REGISTER messages is sent at different times.

**voIpProt.SIP.rport**

0 (default) – The phone does not insert the rport parameter into the Via header of its requests.

1 – The phone inserts the rport parameter, as defined by RFC 3581, into the Via header of its requests.

**voIpProt.SIP.requestURI.E164.addGlobalPrefix**

0 (default) - ‘+’ global prefix is not added to the E.164 user parts in sip: URIs.

1 - ‘+’ global prefix is added to the E.164 user parts in sip: URIs.

**voIpProt.SIP.requestValidation.digest.realm**

Determines the string used for Realm.

PolycomSIP (default)

string

Change causes system to restart or reboot.

**voIpProt.SIP.requestValidation.x.method**

Null (default) - no validation is made.

Source - ensure request is received from an IP address of a server belonging to the set of target registration servers.

digest: challenge requests with digest authentication using the local credentials for the associated registration (line).

both or all: apply both of the above methods.

Change causes system to restart or reboot.

**voIpProt.SIP.requestValidation.x.method**

Null (default) - no validation is made.
Source - ensure request is received from an IP address of a server belonging to the set of target registration servers.
digest: challenge requests with digest authentication using the local credentials for the associated registration (line).
both or all: apply both of the above methods.
Change causes system to restart or reboot.

voIpProt.SIP.requestValidation.x.request
Sets the name of the method for which validation will be applied.
Null (default)
INVITE, ACK, BYE, REGISTER, CANCEL, OPTIONS, INFO, MESSAGE, SUBSCRIBE, NOTIFY, REFER, PRACK, UPDATE
ALL - Phone does not honor the above request methods received from unknown sources.

Note: Intensive request validation may have a negative performance impact due to the additional signaling required in some cases.
Change causes system to restart or reboot.

voIpProt.SIP.requestValidation.x.request.y.event
Determines which events specified with the Event header should be validated; only applicable when voIpProt.SIP.requestValidation.x.request is set to SUBSCRIBE or NOTIFY.
Null (default) - all events will be validated.
A valid string - specified event will be validated.
Change causes system to restart or reboot.

voIpProt.SIP.RFC3261TimerI
0 (default) - Timer I for reliable transport will be fired at five seconds. This parameter does not cause any change for unreliable transport.
1 - Timer I for reliable transport will be fired at zero seconds.

voIpProt.SIP.sendCompactHdrs
0 (default) - SIP header names generated by the phone use the long form, for example From.
1 - SIP header names generated by the phone use the short form, for example \texttt{f}.

\begin{verbatim}
voIpProt.SIP.serverFeatureControl.callRecording
0 (default) - The BroadSoft BroadWorks v20 call recording feature for multiple phones is disabled.
1 - The BroadSoft BroadWorks v20 call recording feature for multiple phones is enabled.
\end{verbatim}

\begin{verbatim}
voIpProt.SIP.serverFeatureControl.cf
0 (default) - The server-based call forwarding is not enabled.
1 - The server-based call forwarding is enabled.
Change causes system to restart or reboot.
\end{verbatim}

\begin{verbatim}
voIpProt.SIP.serverFeatureControl.cf
0 (default) - Disable server-based call forwarding.
1 - Enable server-based call forwarding.
Change causes system to restart or reboot.
\end{verbatim}

\begin{verbatim}
voIpProt.SIP.serverFeatureControl.dnd
0 (default) - Disable server-based DND.
1 - Server-based DND is enabled. Server and local phone DND are synchronized.
\end{verbatim}

\begin{verbatim}
voIpProt.SIP.serverFeatureControl.localProcessing.cf
This parameter depends on the value of voIpProt.SIP.serverFeatureControl.cf.
1 (default) - If set to 1 and voIpProt.SIP.serverFeatureControl.cf is set to 1, the phone and the server perform call forwarding.
0 - If set to 0 and voIpProt.SIP.serverFeatureControl.cf is set to 1, call forwarding is performed on the server side only, and the phone does not perform local call forwarding.
If both voIpProt.SIP.serverFeatureControl.localProcessing.cf and voIpProt.SIP.serverFeatureControl.cf are set to 0, the phone performs local call forwarding and the localProcessing parameter is not used.
\end{verbatim}

\begin{verbatim}
voIpProt.SIP.serverFeatureControl.localProcessing.cf
1 (default) - Allows to use the value for voIpProt.SIP.serverFeatureControl.cf.
0 - Does not use the value for
This parameter depends on the value of voIpProt.SIP.serverFeatureControl.cf.
\end{verbatim}

\begin{verbatim}
voIpProt.SIP.serverFeatureControl.localProcessing.dnd
This parameter depends on the value of voIpProt.SIP.serverFeatureControl.dnd.
If set to 1 (default) and voIpProt.SIP.serverFeatureControl.dnd is set to 1, the phone and the server perform DND.
\end{verbatim}
If set to 0 and `voIpProt.SIP.serverFeatureControl.dnd` is set to 1, DND is performed on the server-side only, and the phone does not perform local DND.

If both `voIpProt.SIP.serverFeatureControl.localProcessing.dnd` and `voIpProt.SIP.serverFeatureControl.dnd` are set to 0, the phone performs local DND and the `localProcessing` parameter is not used.

`voIpProt.SIP.serverFeatureControl.missedCalls`

0 (default) - Server-based missed calls is not enabled.
1 - Server-based missed calls is enabled. The call server has control of missed calls.
Change causes system to restart or reboot.

`voIpProt.SIP.serverFeatureControl.securityClassification`

0 (default) - The visual security classification feature for all lines on a phone is disabled.
1 - The visual security classification feature for all lines on a phone is enabled.
Change causes system to restart or reboot.

`voIpProt.SIP.specialEvent.checkSync.alwaysReboot`

0 (default) - The phone will only reboot if necessary. Many configuration parameter changes can be applied dynamically without the need for a reboot.
1 - The phone always reboot when a NOTIFY message is received from the server with event equal to check-sync even if there has not been a change to software or configuration.

`voIpProt.SIP.specialEvent.checkSync.downloadCallList`

0 (default) - The phone does not download the call list for the logged-in user when a check sync event's NOTIFY message is received from the server.
1 - The phone downloads the call list for the logged-in user when a check sync event's NOTIFY message is received from the server.

`voIpProt.SIP.specialEvent.checkSync.downloadDirectory`

0 (default) - The phone does not download updated directory files after receiving a checksync NOTIFY message.
1 - The phone downloads updated directory files after receiving a checksync NOTIFY message.
1 - The phone downloads the updated directory files along with any software and configuration updates after receiving a checksync NOTIFY message. The files are downloaded when the phone restarts, reboots, or when the phone downloads any software or configuration updates.

Note: The parameter `hotelingMode.type` set to 2 or 3 overrides this parameter.

**voIpProt.SIP.specialEvent.lineSeize.nonStandard**

Controls the response for a line-seize event SUBSCRIBE.

1 (default) - This speeds up the processing of the response for line-seize event.
0 - This will process the response for the line seize event normally

Change causes system to restart or reboot.

**voIpProt.SIP.strictLineSeize**

0 (default) - Dial prompt is provided immediately when you attempt to seize a shared line without waiting for a successful OK from the call server.
1 - The phone is forced to wait for a 200 OK response when receiving a TRYING notify.

**voIpProt.SIP.strictReplacesHeader**

This parameter applies only to directed call pick-up attempts initiated against monitored BLF resources.

1 (default) - The phone requires call-id, to-tag, and from-tag to perform a directed call-pickup when call.directedCallPickupMethod is configured as native.
0 - Call pick-up requires a call id only.

**voIpProt.SIP.strictUserValidation**

0 (default) - The phone is forced to match the user portion of signaling exactly.
1 - The phone will use the first registration if the user part does not match any registration.

**voIpProt.SIP.supportFor100rel**

1 (default) - The phone advertises support for reliable provisional responses in its offers and responses.
0 - The phone will not offer 100rel and will reject offers requiring 100rel.

**voIpProt.SIP.supportFor199**

Determine support for the 199 response code. For details on the 199 response code, see RFC 6228.
0 (Default) - The phone does not support the 199 response code.
1- The phone supports the 199 response code.

**voIpProt.SIP.tcpFastFailover**
0 (default) - A full 32 second RFC compliant timeout is used.
1 - A failover occurs based on the values of `reg.x.server.y.retryMaxCount` and `voIpProt.server.x.retryTimeOut`.

**voIpProt.SIP.tlsDsk.enable**
0 (default) - TLS DSK is disabled.
1 - TLS DSK is enabled.

**voIpProt.SIP.turnOffNonSecureTransport**
0 (default) - Stop listening to port 5060 when using AS-SIP feature is disabled.
1 - Stop listening to port 5060 when using AS-SIP feature is enabled.
Change causes system to restart or reboot.

**voIpProt.SIP.use486forReject**
0 (default) - The phone responds with 603.
1 - The phone responds with 486.

**voIpProt.SIP.useContactInReferTo**
0 (default) - The “To URI” is used in the REFER.
1 - The “Contact URI” is used in the REFER.

**voIpProt.SIP.useLocalTargetUriForLegacyPickup**
1 (default) - The target URI in BLF signaling uses the complete address as provided in the XML dialog document.
0 - Only the user portion of the target URI in the XML dialog document is appended to create the address for pickup or retrieval.

**voIpProt.SIP.useRFC2543hold**
0 (default) - SDP media direction parameters (such as a=sendonly) per RFC 3264 when initiating a call.
1 - the obsolete c=0.0.0.0 RFC2543 technique is used when initiating a call.

**voIpProt.SIP.useRFC2543hold**
0 (default) - SDP media direction parameters (such as a=sendonly) per RFC 3264 when initiating a call.
1 - the obsolete c=0.0.0.0 RFC2543 technique is used when initiating a call.
voIpProt.SIP.useRFC3264HoldOnly

0 (default) - When set to 0, and no media direction is specified, the phone enters backward compatibility mode when negotiating SDP and responds using the c=0.0.0.0 RFC 2543 signaling method.

1 - When set to 1, and no media direction is specified, the phone uses sendrecv compliant with RFC 3264 when negotiating SDP and generates responses containing RFC 3264-compliant media attributes for calls placed on and off hold by either end.

Note: voIpProt.SIP.useSendonlyHold applies only to calls on phones that originate the hold.

voIpProt.SIP.useSendonlyHold

1 (default) - The phone will send a reinvite with a stream mode parameter of "sendonly" when a call is put on hold.

0 - The phone will send a reinvite with a stream mode parameter of "inactive" when a call is put on hold.

Note: The phone will ignore the value of this parameter if set to 1 when the parameter voIpProt.SIP.useRFC2543hold is also set to 1 (default is 0).

voIpProt.SIP.ignoreEntityHost

0 (default) – Doesn’t ignore the host part of the entity received in the XML body of NOTIFY for a dialog event.

1 - Ignores the host part of the entity received in the XML body of NOTIFY for a dialog event.

voIpProt.SIP.forkedRespRecommendedCseq

1 (default) - Generates the RFC compliance Cseq number.

0 - Generates the call specific CSeq number.

Multiple Line Keys Per Registration

You can assign a single registered phone line address to multiple line keys on Poly phones. This feature can be useful for managing a high volume of calls to a single line.

Multiple Line Keys Per Registration Parameter

Use the parameter below to configure this feature. This feature is one of several features associated with Call Appearances.

reg.x.lineKeys

Specify the number of line keys to use for a single registration. The maximum number of line keys you can use per registration depends on your phone model.

1 (default)
Multiple Call Appearances

You can enable each registered phone line to support multiple concurrent calls and have each concurrent call display on the phone's user interface.

For example, with multiple call appearances, users can place one call on hold, switch to another call on the same registered line, and have both calls display on the phone.

This feature is one of several features associated with flexible call appearances. If you assign a registered line to multiple line keys, the default number of concurrent calls applies to all line keys.

Multiple Call Appearance Parameters

Use the parameters in the following list to set the maximum number of concurrent calls per registered line and the default number of calls per line key.

Note that you can set the value for the \texttt{reg.1.callsPerLineKey} parameter to a value higher than 1, for example, 3. After you set the value to 3, for example, you can have three call appearances on line 1. By default, any additional incoming calls are automatically forwarded to voicemail. If you set more than two call appearances, a call appearance counter displays at the top-right corner on the phone.

\textbf{call.callsPerLineKey}

Set the maximum number of concurrent calls per line key. This parameter applies to all registered lines.

\textbf{Note:} This parameter can be overridden by the per-registration parameter \texttt{reg.x.callsPerLineKey}.

12 (default)
1 - 12

\textbf{reg.x.callsPerLineKey}

Set the maximum number of concurrent calls for a single registration \textit{x}. This parameter applies to all line keys using registration \textit{x}. If registration \textit{x} is a shared line, an active call counts as a call appearance on all phones sharing that registration.

This per-registration parameter overrides \texttt{call.callsPerLineKey}.

12 (default)
1 - 12
Flexible Call Appearances

A number of features are associated with flexible call appearances, including Multiple Line Registrations, Multiple Line Keys Per Registration, and Multiple Call Appearances.

Use the following table to understand how you can organize registrations, line keys per registration, and concurrent calls per line key.

Static BLF and EFK are also now supported for FLK.

The following table includes the following types of call appearances:

- Registrations—The maximum number of user registrations
- Line Keys—The maximum number of line keys
- Line Keys Per Registration—The maximum number of line keys per user registration
- Calls Per Line Key—The maximum number of concurrent calls per line key
- Concurrent Calls (including Conference Legs)—The runtime maximum number of concurrent calls, and the number of conference participants minus the conference initiator.

<table>
<thead>
<tr>
<th>Phone Model</th>
<th>Registrations</th>
<th>Line Keys</th>
<th>Line Keys Per Registration</th>
<th>Calls Per Line Key</th>
<th>Concurrent Calls</th>
</tr>
</thead>
<tbody>
<tr>
<td>CCX 400</td>
<td>34</td>
<td>48</td>
<td>48</td>
<td>24</td>
<td>24 (2)</td>
</tr>
<tr>
<td>CCX 500</td>
<td>34</td>
<td>48</td>
<td>48</td>
<td>24</td>
<td>24 (2)</td>
</tr>
<tr>
<td>CCX 600</td>
<td>34</td>
<td>48</td>
<td>48</td>
<td>24</td>
<td>24 (2)</td>
</tr>
<tr>
<td>CCX 700</td>
<td>34</td>
<td>48</td>
<td>48</td>
<td>24</td>
<td>24 (2)</td>
</tr>
</tbody>
</table>

Bridged Line Appearance

Bridged line appearance connects calls and lines to multiple phones.

With bridged line appearance enabled, an active call displays simultaneously on multiple phones in a group. By default, the answering phone has sole access to the incoming call, which is called line seize. If the answering phone places the call on hold, that call becomes available to all phones of that group. All call states—active, inactive, on hold—are displayed on all phones of a group.

**Important:** Shared call appearances and bridged line appearances are similar signaling methods that enable more than one phone to share the same line or registration. The methods you use vary with the SIP call server you are using. In the configuration files, bridged lines are configured by shared line parameters. The barge-in feature is not available with bridged line appearances; it is available only with shared call appearances.
Bridged Line Appearance Signaling

A bridged line is an address of record managed by a server. The server allows multiple endpoints to register locations against the address of record. The phone supports Bridged Line Appearances (BLA) using the SUBSCRIBE-NOTIFY method in the SIP Specific Event Notification framework (RFC 3265). The event used is dialog for bridged line appearance subscribe and notify.

Bridged Line Appearance Parameters

To begin using Bridged Line Appearance, you must get a registered address dedicated for use with your call server provider. This dedicated address must be assigned to a phone line in the `reg.x.address` parameter. Use the parameters in the following list to configure this feature.

**call.shared.disableDivert**

- 1 (default) - Enable the diversion feature for shared lines.
- 0 - Disable the diversion feature for shared lines. Note that this feature is disabled on most call servers.
  
  Change causes system to restart or reboot.

**reg.x.type**

- private (default) - Use standard call signaling.
- shared - Use augment call signaling with call state subscriptions and notifications and use access control for outgoing calls.

**reg.x.thirdPartyName**

- Null (default) - In all other cases.
- string address - This field must match the `reg.x.address` value of the registration which makes up the part of a bridged line appearance (BLA).

**divert.x.sharedDisabled**

- 1 (default) - Disables call diversion features on shared lines.
- 0 - Enables call diversion features on shared lines.

Change causes system to restart or reboot.

**voIpProt.SIP.blaGlareHonorRetryAfter**

Controls the Retry mechanism.

- 1 (default) - The phone honors the Retry-after header on glare and sends NOTIFY with the same state and line-id after the requested time interval.
- 0 - The phone ignores the Retry-after header on glare and immediately sends NOTIFY with the next available line-id.
Voicemail

When you configure phones with a SIP URL that integrates with a voicemail server contact, users receive a visual and audio alert when they have new voicemail messages available on their phone.

Voicemail Parameters

Use the parameters in the following list to configure voicemail and voicemail settings.

**feature.voicemail.enabled**

1 (default) - Enable voicemail.
0 - Disable voicemail.

**msg.mwi.x.callBackMode**

The message retrieval mode and notification for registration x.

registration (default) - The registration places a call to itself (the phone calls itself).
contact - a call is placed to the contact specified by msg.mwi.x.callback.
disabled - Message retrieval and message notification are disabled.

**msg.mwi.x.callback**

The contact to call when retrieving messages for this registration if msg.mwi.x.callBackMode is set to contact.

ASCII encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL (6416 or 6416@polycom.com)
NULL (default)

**msg.mwi.x.subscribe**

Specify the URI of the message center server. ASCII encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL (6416 or 6416@polycom.com)
If non-Null, the phone sends a SUBSCRIBE request to this contact after boot up.
NULL (default)

**mwi.backLight.disable**

Specify if the phone screen backlight illuminates when you receive a new voicemail message.
0 (default) - Disabled
1 - Enabled
Change causes system to restart or reboot.

**up.mwiVisible**

Specify if message waiting indicators (MWI) display or not.
0 (default) - If msg.mwi.x.callBackMode=0, MWI do not display in the message retrieval menus.
1 - MWI display.
Change causes system to restart or reboot.

up.oneTouchVoiceMail
0 (default) - Generic Base Profile
0 (default) - The phone displays a summary page with message counts.
1 - You can call voicemail services directly from the phone, if available on the call server, without displaying the voicemail summary.
Change causes system to restart or reboot.

Local Call Recording
Local call recording enables you to record audio calls to a USB device connected to the phone.
You can play back recorded audio on the phone or using an audio application on the computer. To use this feature, you must enable USB port.
Audio calls are recorded in .wav format and include a date/time stamp. The phone displays the recording time remaining on the attached USB device, and users can browse all recorded files using the phone's menu.

Note: Federal, state, and/or local laws may legally require that you notify some or all of the call parties when a call recording is in progress.

Local Call Recording Parameter
Use the following parameter to configure local call recording.

feature.callRecording.enabled
0 (default) - Disable audio call recording.
1 - Enable audio call recording.
Change causes system to restart or reboot.

Centralized Call Recording
This feature enables users to record audio and video calls and control call recording directly from phones registered with BroadSoft BroadWorks r20 server.
Users can manage recorded audio and video files on a third-party call recording server.
By default, far-side participants are not alerted when calls are being recorded. The BroadWorks r20server provides administrators with the option to enable an announcement to play at the beginning of a call when
a call is being recorded. If a call recorded is in progress when the call is transferred, the recording continues for the new call.

**Note:** You can record calls using a central server or locally using the phone's USB call recording feature - you cannot use both at the same time. By default, both features are disabled. If you enable one call recording feature, ensure that the other is disabled. Use either centralized or the local call recording; do not use both.

### Centralized Call Recording Parameters

You must enable this feature on the BroadSoft BroadWorks r20 server and on the phones using the configuration parameters listed in the following list.

On the BroadSoft server, assign phone users one of several call recording modes listed in

```
voIpProt.SIP.serverFeatureControl.callRecording
```

- **0** (default) - The BroadSoft BroadWorks v20 call recording feature for multiple phones is disabled.
- **1** - The BroadSoft BroadWorks v20 call recording feature for multiple phones is enabled.

  Change causes system to restart or reboot.

```
reg.x.serverFeatureControl.callRecording
```

- **1** (default) - BroadSoft BroadWorks v20 call recording feature for individual phone lines is enabled.
- **0** - BroadSoft BroadWorks v20 call recording feature for individual phone lines is disabled.

### Call Recording Modes

Set the call recording modes on the BroadSoft BroadWorks R20 server using the following call recording modes:

- **Never Mode** – Call recording is never initiated and the phone never displays call recording soft keys.

- **Always Mode** – The entire incoming or outgoing call is recorded and no control options are available to users. During active calls, the phone displays a Record symbol. Call recording stops when the call ends and the call is stored on the server.

- **Always with Pause/Resume Support Mode** – Call recording starts automatically when the call connects and the Pause and Resume soft keys are available. The phone display indicates the status of the call recording state. Call recording stops when the call ends and the recorded part of the call is stored on the server.

- **On Demand Mode** – Call recording starts on the server when the call connects, but the recorded file is not saved until the user initiates the recording. When the user presses the Start soft key, the recording is saved to the server and the phone displays the Pause and Resume soft keys.

- **On Demand Mode with User-Initiated Start Mode** – Call recording does not begin automatically and a Record soft key displays. If users want to record an active call, they need to press Record > Start to start recording and save the recording to the server. While recording, the phone displays the Pause, Resume, and Stop soft keys.
• **Recording two separate calls and creating a conference** – This mode enables users to record two participants as separate call sessions when connected in a conference call. The server stores the conference call as two separate recording sessions.

**Busy Lamp Field (BLF)**

The busy lamp field (BLF)/attendant console feature enhances support for phone-based monitoring.

The Busy Lamp Field (BLF) feature enables the following functions for users:

- Monitor the status of lines on remote phones
- Display remote party information
- Answer incoming calls to remote phones (called directed call pickup)
- Park and retrieve calls

When BLF is enabled, a BLF line key icon displays on the phone screen for users monitoring remote phones. The BLF line key displayed indicates that BLF related features are available.

**BLF Icons**

The following table shows the BLF key icons that display on the phone.

<table>
<thead>
<tr>
<th>States</th>
<th>Line Icons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monitored line is idle</td>
<td>[Icon]</td>
</tr>
<tr>
<td>Monitored line is busy</td>
<td>[Icon]</td>
</tr>
<tr>
<td>Monitored line is in hold</td>
<td>[Icon]</td>
</tr>
<tr>
<td>Monitored line is unregistered</td>
<td>[Icon]</td>
</tr>
</tbody>
</table>

**Busy Lamp Field Actions**

When a monitored Busy Lamp Field (BLF) resource indicates hold, the short press action of the BLF line key picks up the call.

When the parameter `attendant.displayHoldState.enable` is enabled, the phone picks up the latest held call if there are multiple hold calls on BLF line.

If the parameter is disabled, the monitored user call should be on hold. When the monitoring user short presses on the BLF line keys, a new call will be initiated to the BLF line.

**BLF Feature Options**

The BLF feature must be supported by a call server and the specific functions vary with the call server you use.

You may need to consult your SIP server partner or Poly channel partner to find out how to configure BLF feature options.
You can configure the following feature options for BLF:

- Line key labels
- Enhanced feature keys
- Call appearances display
- Call waiting audio notifications
- Caller ID information display
- One-touch call park and retrieve
- One-touch directed call pickup

**BLF Configuration Methods**

Typically, call servers support one of two methods of BLF configuration.

Using the first method, you subscribe to a BLF resource list set up on your call server. Using the second method, you enter BLF resources to a configuration file and the call server directs the requests to those BLF resources. If you are unsure which method to use, consult your SIP server partner or Poly Channel partner. This section shows you how to set up BLF using both methods.

When using BLF with a call server, the initial BLF subscription can receive large responses as the number of monitored resources increases. To ensure transmission, Poly recommends using Transmission Control Protocol (TCP) for BLF either by changing all SIP services to TCP or by adding the TCP transport attribute to your `attendant.uri` parameter. For example, configure `attendant.uri=1234blf@example;transport=tcp`.

**BLF Resource List Subscription on a Call Server**

To subscribe to a BLF list on a call server, you must access the call server and set up a list of monitored resources.

The call server provides you with an address for that BLF resource list. To subscribe to that list, enter the address and any other information specific to your call server in the `attendant.uri` parameter.

**BLF Resource Specification in the Configuration File**

To specify BLF resources, enter the address (phone number) of the BLF resource of the monitored contact, the label that displays beside the line key on the phone, and the type of resource being monitored.

Multiple registrations are available for a single SIP server. Your call server must support dialog even package defined in RFC 4235 in order to configure BLF using this method. In the following example, the phone is monitoring Craig Blunt and Lucy Patterson.

Specifying the type of monitored resource as `normal` or `automata` changes the default actions of key presses. Enter `normal` as the resource type if the monitored resource type is a phone and `automata` as the resource type if the monitored resource type is, for example, a call orbit. If you select `normal`, pressing the BLF line key places an active call on hold before dialing the selected BLF phone. If you select `automata`, pressing the BLF line key immediately transfers active calls to that resource.

**Busy Lamp Field Configuration Parameters**

The maximum number of BLF entries for phones is 50.

In the following list, x in a parameter is the number of the BLF entry in the list. If you are using static BLF, you need to configure the number of each entry.
**attendant.behaviors.automata.pickupOnBusy**

Set to allow an automata resource (static BLF) pickup on a busy BLF Resource.

1 (default) - Allows pick up on a Busy Lamp Field resource.
0 - Doesn’t allow pick up on a Busy Lamp Field resource.

**attendant.behaviors.display.remoteCallerID.automata**

These parameters depend on the value set for the parameter

attendant.resourceList.x.type . If the parameter

attendant.resourceList.x.type is set to automata, use the parameter

attendant.behaviors.display.remoteCallerID.automata.

1 (default) - Automata remote party caller ID information is presented to the attendant.
0 - The string unknown is substituted for both name and number information.

**attendant.behaviors.display.remoteCallerID.normal**

These parameters depend on the value set for the parameter

attendant.resourceList.x.type . If the parameter

attendant.resourceList.x.type is set to normal, use the parameter

attendant.behaviors.display.remoteCallerID.normal.

1 (default) - Normal remote party caller ID information is presented to the attendant.
0 - The string unknown is substituted for both name and number information.

**attendant.behaviors.display.spontaneousCallAppearances.automata**

0 (default) - The call appearance is not spontaneously presented to the attendant. The information displayed after a press and hold of a resource’s line key is unchanged by this parameter. Note that the values of these call appearance parameters depend on the values applied to attendant.resourceList.x.type.

When this parameter is set to 0, the ringtone 'Ring Splash' does not play when

attendant.ringType=ringer14.

1 - The normal or automatic call appearance is spontaneously presented to the attendant when calls are alerting on a monitored resource (and a ring tone is played).

**attendant.behaviors.display.spontaneousCallAppearances.normal**

1 (default) - The normal or automatic call appearance is spontaneously presented to the attendant when calls are alerting on a monitored resource (and a ring tone is played).

0 - The call appearance is not spontaneously presented to the attendant. The information displayed after a press and hold of a resource’s line key is unchanged by this parameter. Note that the values of these call appearance parameters depend on the values applied to attendant.resourceList.x.type.

When this parameter is set to 0, the ringtone 'Ring Splash' does not play when

attendant.ringType=ringer14.

**attendant.behaviours.display.spontaneousCallAppearances.automata**

Specifies how call appearances display on the attendant phone.
0 (default) - The call appearance is not spontaneously presented to the attendant. The information displayed after a press and hold of a resource’s line key is unchanged by this parameter.

1 - The automata call appearance is spontaneously presented to the attendant when calls are alerting on a monitored resource (and a ring tone is played). Note that the values of these call appearance parameters depend on the values applied to 
`attendant.resourceList.x.type`.

**attendant.behaviours.display.spontaneousCallAppearances.normal**

Specifies how call appearances display on the attendant phone.

1 (default) - The normal call appearance is spontaneously presented to the attendant when calls are alerting on a monitored resource (and a ring tone is played).

0 - The call appearance is not spontaneously presented to the attendant. The information displayed after a press and hold of a resource’s line key is unchanged by this parameter.

**Note:** That the values of these call appearance parameters depend on the values applied to 
`attendant.resourceList.x.type`.

**attendant.x.display.spontaneousCallAppearances**

Specifies spontaneous call appearance property for any BLF incoming call.

This parameter will override the phone level configuration parameters 
`attendant.behaviors.display.spontaneousCallAppearances.normal` and 
`attendant.behaviors.display.spontaneousCallAppearances.automata` to show 
or hide the call appearance for any BLF incoming call.

Auto (default) – This value will use phone-level configuration depending on the BLF value set of 
parameters.

Show – This value will override phone-level configuration and will show the call appearance.

Hide – This value will override phone-level configuration and will hide the call appearance.

**attendant.callWaiting.enable**

0 (default) - The phone does not generate acoustic indication of call waiting for attendant calls monitored by BLF.

1 - The phone generates an acoustic indication of call waiting for attendant calls monitored by BLF.

**attendant.callWaiting.ring**

This parameter is valid only if `attendant.callWaiting.enable` is set to 1. Specifies the ring type to be used for notifying an attendant call if there is an active call already present on the phone.

Silent - No acoustic indication is provided.

beep - Beep tone is played when there is an active call on the phone and an attendant call is received.

ring - Ring tone configured in `attendant.ringType` is used to alert the user when there is an active call on the phone and an attendant call is received.
attendant.reg

Specifies an index number for the BLF resource. The index of the registration is used to send a SUBSCRIBE to the list SIP URI specified in attendant.uri. For example, attendant.reg = 2 means the second registration is used.

1 (default)
Permitted value is any positive integer.

attendant.resourceList.x.address

The user referenced by attendant.reg="" subscribes to this URI for dialog. If a user part is present, the phone subscribes to a sip URI constructed from the user part and domain of the user referenced by attendant.reg. Transport for BLF subscriptions may be modified by including a transport parameter into the subscription address. For example: sip: blf12345@domain.com;transport=tcp

Permitted value is a string that constitutes a valid SIP URI (sip: 6416@polycom.com) or contains the user part of a SIP URI (6416).

Null (default)

attendant.resourceList.x.bargeInMode

Enable or disable barge-in and choose the default barge-in mode. This parameter applies to the Alcatel-Lucent CTS only.

Null (default) - If no value is entered, the Barge In feature is disabled.

All - Press and hold the BLF line to display all barge-in options.

Quick press to barge-in as Normal.

Normal - Barge-in plays an audio tone to indicate the arrival of a new participant to the call and all call participants can interact.

Listen - The user barging in can listen on the call only. Their outbound audio is not transmitted to either party.

Whisper - The user barging in can hear all parties but their audio is only transmitted to the user they are monitoring.

attendant.resourceList.x.callAddress

Use this parameter when the call signaling address for the BLF line is different than the address set by attendant.resourceList.x.address.

Null (default)
Maximum 255 characters

attendant.resourceList.x.label

The text label displays adjacent to the associated line key. If set to Null, the label is derived from the user part of attendant.resourceList.x.address.

Null (default)
Permitted value is a UTF-8 encoded string.
attendant.resourceList.x.proceedingIsRecipient

A flag to determine if pressing the associated line key for the monitored user picks up the call.
1 - If the call server does not support inclusion of the direction attribute in its dialog XML.
0 (default)

attendant.resourceList.x.requestSilentBargeIn

0 (default) - A tone plays when a contact barges in on a call.
1 - No tone is played when a contact barges in on a call.

attendant.resourceList.x.type

The type of resource being monitored and the default action to perform when pressing the line key adjacent to monitored user x.

normal (default) - The default action is to initiate a call if the user is idle or busy and to perform a directed call pickup if the user is ringing. Any active calls are first placed on hold.

Note: The value normal applies the call appearance setting
attendant.behaviors.display.*.normal.

automata - The default action is to perform a park/blind transfer of any currently active call. If there is no active call and the monitored user is ringing/busy, an attempt to perform a directed call pickup/park retrieval is made.

Note: That the value automata applies the call appearance setting
attendant.behaviors.display.*.automata=0.

attendant.restrictPickup

0 (default) - The attendant can pick up calls to monitored users while they show as ringing.
1 - The attendant cannot pick up the monitored call.

attendant.ringType

The ringtone that plays when a BLF dialog is in the offering state.

ringer1 (default)
ringer1 - ringer 24

attendant.uri

The list SIP URI on the server. If this is just a user part, the URI is constructed with the server hostname/IP.

Note: If this parameter is set, then the individually addressed users configured by
attendant.resourceList and attendant.behaviors are ignored.

Null (default)
Strings are permitted.
**call.directedCallPickupMethod**

Specifies how the phone performs a directed call pick-up from a BLF contact.
- **legacy (default)** - Indicates that the phone uses the method specified in `call.directedCallPickupString`.
- **native** - Indicates that the phone uses a native protocol method (in this case SIP INVITE with the Replaces header).

**call.directedCallPickupString**

The star code to initiate a directed call pickup.
- `*97` (default)

*Note:* The default value supports the BroadWorks calls server only. You must change the value if your organization uses a different call server.

**call.parkedCallRetrieveMethod**

The method the phone uses to retrieve a BLF resource's call which has dialog state confirmed.
- **legacy (default)** - Indicates that the phone uses the method specified in `call.parkedCallRetrieveString`.
- **native** - Indicates that the phone uses a native protocol method (in this case SIP INVITE with the Replaces header).

**call.parkedCallRetrieveString**

The star code that initiates retrieval of a parked call.
- Null (default)

Permitted values are star codes.

**voipPort.SIP.useCompleteUriForRetrieve**

1 (default) - The target URI in BLF signaling uses the complete address as provided in the XML dialog document.
0 - Only the user portion of the target URI in the XML dialog document is used and the current registrar's domain is appended to create the address for retrieval.

**voIpProt.SIP.strictReplacesHeader**

This parameter applies only to directed call pick-up attempts initiated against monitored BLF resources.
1 (default) - The phone requires call-id, to-tag, and from-tag to perform a directed call-pickup when `call.directedCallPickupMethod` is configured as native.
0 - Call pick-up requires a call id only.

**voIpProt.SIP.useLocalTargetUriforLegacyPickup**

1 (default) - The target URI in BLF signaling uses the complete address as provided in the XML dialog document.
0 - Only the user portion of the target URI in the XML dialog document is used and the current registrar's domain is appended to create the address for pickup or retrieval.

**attendant.callAction**

Specify the call action behavior for an Active call.

- **Dial-Pick up (default)** – An active call goes on hold and dials to monitor line or picks the incoming call on monitor line when you short press the monitored line keys.
- **Blind** – Blind transfer an active call on the monitored line keys.
- **Park** – Parks an active call on the monitored line keys. If there is already a parked call on a monitored line then it will retrieve the parked call.

**attendant.callActionMenu.enabled**

This parameter is configured to get the **Attendant Call Action** menu on the phone when you configure dynamic BLF on the phone.

- 0 (default) – **Attendant Call Action** menu will not appear on the phone.
- 1 - **Attendant Call Action** menu will appear on the phone.

**attendant.displayHoldState.enable**

Specifies the control of the display on the phone for BLF hold state.

- 0 (default) - The phone displays a busy state.
- 1 - The phone displays a hold state.

*Note : This parameter is only applicable to static BLF*

**attendant.resourceList.x.hold.ringer**

The ringtone that plays on the phone when BLF is in a hold state.

- The parameter depends on the value set for the parameter attendant.displayHoldState.enable.
  - If the parameter attendant.displayHoldState.enable is set to 1, use the parameter attendant.resourceList.x.hold.ringer

- **Triplet (default)** – Specifies the ringtone name for the parameter ringer11.

- Ringtone for BLF Hold should play for only 10 sec.

**attendant.resourceList.x.display.spontaneousCallAppearances**

This parameter is applicable to Static BLF.

- Specifies spontaneous call appearance property for an incoming call.

- This parameter will override the phone level configuration parameters
  attendant.behaviors.display.spontaneousCallAppearances.normal and
  attendant.behaviors.display.spontaneousCallAppearances.automata to show
  or hide the call appearance property for BLF incoming call based on the resource type.

- **Auto (default)** – This value will use phone-level configuration depending on the BLF resource type.

- **Show** - This value will override phone-level configuration and show the call appearance.

- **Hide** - This value will override phone-level configuration and hide the call appearance.
Note: Existing BLF ringtone will not stop, if new BLF call comes.

attendant.resourceList.x.ringType
This parameter is applicable to Static BLF.
Specifies incoming ringtone for each static BLF line
defaultAll (default) – Specifies the ringtone type ring for the ringtone name.
ringer1 - ringer 24.
If no ringtone is configured for any static BLF line, then phone level incoming ringtone defined with attendant.ringType parameter will be played.

Key System Emulation
Key System Emulation (KSE) allows one-touch call park and call retrieve from any phone within the user group.
A user group is a set of users defined by the administrator. Each user of the user group can monitor the same group of line keys.
You must enable Busy Lamp Field (BLF) and Enhanced Call Park features on the phone for KSE to work seamlessly. BLF and KSE are mutually exclusive. If you enable KSE, BLF is no longer available to monitor calls.
Key System Emulation includes the following behavior:
• An audio notification plays on the phones in the user group when someone parks a call.
• A reminder tone continuously plays after a designated time interval if no one answers the call.
• There are no audio and reminder notifications for a self-parked call.
• The LED patterns and the line icons for a self-parked call are different from a call parked by other users in the group. This helps to differentiate between a self-parked call and a remote-parked call.
• The LED indicator turns solid red for a self-parked call and turns blinking red for a remote-parked call.

Note: Key System Emulation is applicable to only the BroadSoft call control platform

Key System Emulation Parameters
Use the following parameters to configure the KSE feature on supported phones.

attendant.keylineEmulation.enabled
0 (default) - Disables the KSE feature.
1 - Enables the KSE feature.

attendant.keylineEmulation.showParkedCallerId
1 (default) - The display name of the parked caller (if available) is shown for a line whenever a call is parked.
0 - The display name of the parked caller is generated from BLF dialog resource list.

**feature.enhancedCallPark.allowBLFAudioNotification**

Allow call park audio notification on BLF monitored lines.

0 (default) - Disabled

1 - Enabled

This parameter is applicable only if KSE is enabled.

**attendant.callParkBLFReminder.StartDelay**

Time in seconds before the first reminder tone is played.

0 (default) - No reminder tone is played for calls parked by remote phones.

0 - 3600

This parameter is applicable only if KSE is enabled.

**attendant.callParkBLFReminder.RepeatTime**

Time in seconds between two reminder tones.

0 (default) - No repeat reminder tone is played.

*(When attendant.callParkBLFReminder.StartDelay parameter is not set to 0 and attendant.callParkBLFReminder.RepeatTime parameter is set to 0, a single start reminder tone is played.)*

0-3600

This parameter is applicable only if KSE is enabled.

**Configuring Key System Emulation**

The following sample configuration provides an example of how to set up the Key System Emulation (KSE) feature.

```plaintext
attendant.uri="4455@rxx.polycom.com"
attendant.CallAction="Park"
attendant.callActionMenu.enabled="1"
attendant.reg="1"
reg.1.address="4022"
reg.1.server.1.address="rxx.polycom.com"
feature.enhancedCallPark.allowBLFAudioNotification="1"
attendant.keylineEmulation.enabled="1"
attendant.callParkBLFReminder.StartDelay="10"
attendant.callParkBLFReminder.RepeatTime="5"
se.pat.misc.callParkBLFReminderTone.inst.1.type="chord"
se.pat.misc.callParkBLFReminderTone.inst.1.value="cs4"
se.pat.misc.callParkBLFReminderTone.inst.1.param="0"
se.pat.misc.callParkBLFReminderTone.inst.1.atten="0"
se.pat.misc.callParkBLFAudioNotification.inst.1.type="chord"
se.pat.misc.callParkBLFAudioNotification.inst.1.value="cs4"
se.pat.misc.callParkBLFAudioNotification.inst.1.param="0"
se.pat.misc.callParkBLFAudioNotification.inst.1.atten="0"
```
Instant Messaging
Send and receive instant text messages through your phone.

Note: Support for Instant Messaging varies by call server. Consult your SIP server partner to find out if this feature is supported.

When instant messaging is enabled, the phone’s message waiting indicator (MWI) visually alerts users new instant messages. You can also set audio alerts.

Instant Messaging Parameter

feature.messaging.enabled
0 (default) - Disable instant messaging.
1 - Enable instant messaging.
Change causes system to restart or reboot.

Local and Centralized Conference Calls
You can set up local or centralized audio and video conferences.

Local conferences require a host phone to process the audio and video of all parties. Alternatively, users can use an external audio bridge, available via a central server, to create a centralized conference call. All Poly phones support local- and server-based centralized conferencing. Poly recommends using centralized conferencing for conferences with four or more parties. The availability of centralized conferencing and features can vary by the call platform you use.

Local and Centralized Conference Call Parameters
The following list includes available call management parameters.

You can specify whether, when the host of a three-party local conference leaves the conference, the other two parties remain connected or disconnected. If you want the other two parties remain connected, the phone performs a transfer to keep the remaining parties connected. If the host of four-party local conference leaves the conference, all parties are disconnected and the conference call ends. If the host of a centralized conference leaves the conference, each remaining party remains connected. For more ways to manage conference calls, see Conference Management.

call.localConferenceCallHold
0 (default) - The host cannot place parties on hold.
1 - During a conference call, the host can place all parties or only the host on hold.
call.transferOnConferenceEnd

1 (default) - After the conference host exits the conference, the remaining parties can continue.
0 - After the conference host exits the conference, all parties are exited and the conference ends.

call.singleKeyPressConference

Specify whether or not all parties hear sound effects while setting up a conference.
0 (default) - Phone sound effects are heard only by the conference initiator.
1 - A conference is initiated when a user presses Conference the first time. Also, all sound effects (dial tone, DTMF tone while dialing and ringing back) are heard by all participants in the conference.

voIpProt.SIP.conference.address

Null (default) - Conferences are set up on the phone locally.
String 128 max characters - Enter a conference address. Conferences are set up by the server using the conferencing agent specified by this address. Acceptable values depend on the conferencing server implementation policy.

Conference Management

This feature enables users to add, hold, mute, share video with, and remove conference participants, as well as obtain additional information about participants.

When you enable conference management, a Manage softkey displays on the phone during a conference. The Manage softkey provides access to conference management options.

Conference Management Parameter

Use the parameter in the following list to configure the conference management feature.

feature.nWayConference.enabled

0 - Users can hold three-way conferences but conference management options are not available.
1 - Users can hold conferences with the maximum number of parties, and the conference management options display to enable users to add, hold, mute, and remove participants.

Local Digit Map

The local digit map feature allows the phone to automatically call a dialed number you configure.

Digit maps are defined by a single string or a list of strings. If a dialed number matches any string of a digit map, the call is automatically placed. If a dialed number matches no string—an impossible match—you can specify the phone’s behavior. If a number ends with #, you can specify the phone’s behavior, called trailing # behavior. You can also specify the digit map timeout, the period of time after you dial a
number that the call is placed. The configuration syntax of the digit map is based on recommendations in section 2.1.5 of RFC 3435.

**Local Digit Maps Parameters**

Use the following parameters to configure the local digit map.

**dialplan.applyToCallListDial**

Choose whether the dial plan applies to numbers dialed from the received call list or missed call list, including sub-menus.

1 (default) - Enabled
0 - Disabled

Change causes system to restart or reboot.

**dialplan.applyToDirectoryDial**

Generic Base Profile – 0 (default)

0 - The dial plan is not applied to numbers dialed from the directory or speed dial, including auto-call contact numbers.
1 - The dial plan is applied to numbers dialed from the directory or speed dial, including auto-call contact numbers.

Change causes system to restart or reboot.

**dialplan.applyToForward**

Generic Base Profile – 0 (default)

0 - The dial plan does not apply to forwarded calls.
1 - The dial plan applies to forwarded calls.

Change causes system to restart or reboot.

**dialplan.applyToTelUriDial**

Choose whether the dial plan applies to URI dialing.

1 (default) - Enabled
0 - Disabled

Change causes system to restart or reboot.

**dialplan.applyToUserDial**

Choose whether the dial plan applies to calls placed when the user presses Dial.

1 (default) - Enabled
0 - Disabled

Change causes system to restart or reboot.
**dialplan.applyToUserSend**
Choose whether the dial plan applies to calls placed when the user presses Send.

1 (default) - Enabled
0 - Disabled

Change causes system to restart or reboot.

**dialplan.conflictMatchHandling**
Selects the dialplan based on more than one match with the least timeout.

0 (default for Generic Profile) - Disabled
1 - Enabled

**dialplan.digitmap.timeOut**
Specify a timeout in seconds for each segment of the digit map using a string of positive integers separated by a vertical bar (|). After a user presses a key, the phone waits this many seconds before matching the digits to a dial plan and dialing the call.

(Default) 3 | 3 | 3 | 3 | 3 | 3

If there are more digit maps than timeout values, the default value 3 is used. If there are more timeout values than digit maps, the extra timeout values are ignored.

Change causes system to restart or reboot.

**dialplan.digitmap**
Specify the digit map used for the dial plan using a string compatible with the digit map feature of MGCP described in 2.1.5 of RFC 3435. This parameter enables the phone to automatically initiate calls to numbers that match a digit map pattern.

Generic Base Profile (default) –

```
```

The string is limited to 2560 bytes and 100 segments of 64 bytes, and the following characters are allowed in the digit map.

- A comma (,), which turns dial tone back on.
- A plus sign (+) is allowed as a valid digit.
- The extension letter 'R' indicates replaced string.
- The extension letter 'Pn' indicates precedence, where 'n' range is 1-9.
  
  1 - Low precedence
  9 - High precedence

Change causes system to restart or reboot.

**dialplan.filterNonDigitUriUsers**
Determine whether to filter out (+) from the dial plan.
Call Controls

0 (default) - Disabled
1 - Enabled
Change causes system to restart or reboot.

dialplan.impossibleMatchHandling
0 — The digits entered up to and including the point an impossible match occurred are sent to
the server immediately.
1 — The phone gives a reorder tone.
2 — Users can accumulate digits and dispatch the call manually by pressing Send.
3 (default) — No digits are sent to the call server until the timeout is configured by
dialplan.impossibleMatchHandling.timeout parameter.
If a call orbit number begins with a pound (#) or asterisk (*), you need to set the value to 2 to
retrieve the call using off-hook dialing.
Change causes system to restart or reboot.

dialplan.removeEndOfDial
Sets if the trailing # is stripped from the digits sent out.
1 (default) - Enabled
0 - Disabled
Change causes system to restart or reboot.

dialplan.routing.emergency.outboundIdentity
Choose how your phone is identified when you place an emergency call.
NULL (default)
10-25 digit number
SIP
TEL URI
If using a URI, the full URI is included verbatim in the P-A-I header. For example:
• dialplan.routing.emergency.outboundIdentity = 5551238000
• dialplan.routing.emergency.outboundIdentity=sip:john@emergency.com
• dialplan.routing.emergency.outboundIdentity = tel:+16045558000

dialplan.routing.emergency.preferredSource
Set the precedence of the source of emergency outbound identities.
ELIN (default)— the outbound identity used in the SIP P-Asserted-Identity header is taken from
the network using an LLDP-MED Emergency Location Identifier Number (ELIN).
Config— the parameter dialplan.routing.emergency.outboundIdentity has
priority when enabled, and the LLDP-MED ELIN value is used if
dialplan.routing.emergency.outboundIdentity is NULL.
**dialplan.routing.emergency.x.description**

Set the label or description for the emergency contact address.

`x=1`: Emergency, `Others`: NULL (default)

String

`x` is the index of the emergency entry description where `x` must use sequential numbering starting at 1.

Change causes system to restart or reboot.

**dialplan.routing.emergency.x.server.y**

Set the emergency server to use for emergency routing

(`dialplan.routing.server.x.address` where `x` is the index).

`x=1`: 1, `Others`: Null (default)

Positive integer

`x` is the index of the emergency entry and `y` is the index of the server associated with emergency entry `x`. For each emergency entry (`x`), one or more server entries (`x,y`) can be configured. `x` and `y` must both use sequential numbering starting at 1.

Change causes system to restart or reboot.

**dialplan.routing.emergency.x.value**

Set the emergency URL values that should be watched for. When the user dials one of the URLs, the call is directed to the emergency server defined by `dialplan.routing.server.x.address`.

`x=15`: 911, `Others`: Null (default)

SIP URL (single entry)

`x` is the index of the emergency entry description where `x` must use sequential numbering starting at 15.

**dialplan.routing.server.x.address**

Set the IP address or hostname of a SIP server to use for routing calls. Multiple servers can be listed starting with `x=1` to 3 for fault tolerance.

Null (default)

IP address

Hostname

Blind transfer for 911 or other emergency calls may not work if registration and emergency servers are different entities.

Change causes system to restart or reboot.

**dialplan.routing.server.x.port**

Set the port of a SIP server to use for routing calls.

5060 (default)

1 to 65535
dialplan.routing.server.x.transport
Set the DNS lookup of the first server to use and dialed if there is a conflict with other servers.
  DNSnaptr (default)
  TCPpreferred
  UDPOnly
  TLS
  TCPOnly
  For example, if dialplan.routing.server.1.transport = "UDPOnly" and
dialplan.routing.server.2.transport = "TLS", then UDPOnly is used.

Change causes system to restart or reboot.

dialplan.userDial.timeOut
Specify the time in seconds that the phone waits before dialing a number entered while the
phone is on hook.
  Generic Base Profile (default) – 0
  0-99 seconds
  You can apply dialplan.userDial.timeOut only when its value is lower than
up.IdleTimeOut.

OpenSIP Digit Map
If you are using a list of strings, each string in the list can be specified as a set of digits or timers, or as an
expression which the gateway uses to find the shortest possible match.

In addition, the digit map feature allows SIP URI dialing to match the URIs based on dial plan.

The following is a list of digit map string rules for open SIP environments.

• The following letters are case sensitive: x, T, R, S, and H.
• You must use only *, #, +, or 0-9 between the second and third R.
• If a digit map does not comply, it is not included in the digit plan as a valid map. That is, no match is
  made.
• There is no limit to the number of R triplet sets in a digit map. However, a digit map that contains
  less than a full number of triplet sets (for example, a total of 2 Rs or 5 Rs) is considered an invalid
digit map.
• Digit map extension letter R indicates that certain matched strings are replaced. Using an RRR
  syntax, you can replace the digits between the first two Rs with the digits between the last two Rs.
  For example, R555R604R would replace 555 with 604. Digit map timer letter T indicates a timer
  expiry. Digit map protocol letters S and H indicate the protocol to use when placing a call.
• If you use T in the left part of RRR's syntax, the digit map will not work. For example, R0TR322R
  will not work.

The following examples illustrate the semantics of the syntax:

• R9R604Rxxxxxx-Replaces 9 with 604.
• xxR601R600Rxx-When applied to 1160122 gives 1160022.
• R9RRxxxxxxx-Remove 9 at the beginning of the dialed number (replace 9 with nothing).
  ◦ For example, if you dial 914539400, the first 9 is removed when the call is placed.
• RR604Rxxxxxxx-Prepend 604 to all seven-digit numbers (replace nothing with 604).
  ◦ For example, if you dial 4539400, 604 is added to the front of the number, so a call to
    6044539400 is placed.
• xR60xR600Rxxxxxxx-Replace any 60x with 600 in the middle of the dialed number that matches.
  For example, if you dial 16092345678, a call is placed to 16002345678.
• 911xxx.T-A period (.) that matches an arbitrary number, including zero, of occurrences of the
  preceding construct. For example:
  ◦ 911123 with waiting time to comply with T is a match
  ◦ 9111234 with waiting time to comply with T is a match
  ◦ 91112345 with waiting time to comply with T is a match and the number can grow indefinitely
given that pressing the next digit takes less than T.
• sip:\764xxxxxRR@registrar.polycomcsn.comR-appends
  @registrar.polycomcsn.com to any URI calls matching with "764xxxxx".
  For example, if you make a SIP URI call with 76412345 then @registrar.polycomcsn.com is
  appended to the string such that the SIP URI call INVITE becomes sip::
  76412345@vc.polycom.com. Here, @domain string is required only for SIP URI calls from
  unregistered lines.
• sip:\xxxx\@registrar\.polycomcsn\.com - This will match with any four digit URI calls
  having the domain @registrar.polycomcsn.com.
  For example, if you configure three lines and has dial plan based line switching enabled. Now, if the
  third line's dial plan has sip:\xxxx\@registrar\.polycomcsn\.com then call will be initiated
  from the third line if user dial 1234@registrar.polycomcsn.com because it matches with the
  third line's dial plan.

Generating Secondary Dial Tone with Digit Maps
Enter a short description.
You can dial seven-digit numbers after dialing "8" as shown next in the example rule 8,[2-9]xxxxxXT:
By adding the digit "8", the dial tone plays again, and users can complete the remaining seven-digit
number. In this example, if users also have a 4-digit extension that begins with "8", then users will hear
dial tone after the first "8" was dialed because "8" matches the "8" in the digit map.
If you want to generate dial tone without the need to send the "8", replace one string with another using
the special character "R" as shown next in the rule R8RR. In the following example, replace "8" with an
empty string to dial the seven-digit number:

Enhanced 911 (E.911)
This E.911 feature allows you to configure one of three sources the phone obtains location information from:
Call Controls

- LLDP-MED
- DHCP via option 99
- LIS compliant with RFC 5985

Configuring the source of location information allows the phone to share its location details in the invite sent when a 911 call is made to ensure the 911 operator dispatches emergency services to the correct address.

**Enhanced 911 (E.911) Parameters**

Use the following parameters to configure E.911.

**feature.E911.locationInfoSchema**

- HYBRID (default) - SIP invites use an XML schema as per the RFC4119 and RFC5139 standards.
- RFC 4119 - SIP invites use an XML schema as per the RFC4119 standards.
- RFC5139 - SIP invites use an XML schema as per the RFC5139 standards.

**feature.E911.HELD.server**

NULL (default)

Set the IP address or hostname of the Location Information Server (LIS) address. For example, host.domain.com or https://xxx.xxx.xxx.xxx.

0 - 255

**feature.E911.HELD.username**

NULL (default)

Set the user name used to authenticate to the LIS.

**feature.E911.HELD.password**

NULL (default)

Set the password used to authenticate to the Location Information Server.

0 - 255

**feature.E911.HELD.identity**

Set the vendor-specific element to include in a location request message. For example, 'companyID'.

NULL (default)

String 255 character max

**feature.E911.HELD.identityValue**

Set the value for the vendor-specific element to include in a location request message.

NULL (default)
feature.E911.locationRetryTimer
Specify the retry timeout value in seconds for the location request sent to the Location Information Server (LIS).
The phone does not retry after receiving location information received through the LIS.
60 seconds (default)
60 - 86400 seconds

feature.E911.HELD.nai.enable
0 (default) – The NAI is omitted as a device identity in the location request sent to the LIS.
1 - The NAI is included as a device identity in the location request sent to the LIS.

locInfo.source
Specify the source of phone location information. This parameter is useful for locating a phone in environments that have multiple sources of location information.
LLDP (default for Generic Base Profile) – Use the network switch as the source of location information.
LIS – Use the location information server as the source of location information. Generic Base Profile only.
DHCP – Use DHCP as the source of location information. Generic Base Profile only.
If location information is not available from a default or configured source, the fallback priority is as follows:
Generic Base Profile: No fallback supported for Generic Base Profile

locInfo.x.label
Enter a label for the location.
Null (default)

locInfo.x.country
Enter the country where the phone is located.
Null (default)

locInfo.x.A1
Enter the national subdivision where the phone is located. For example, a state or province.
Null (default)

locInfo.x.A3
Enter the city where the phone is located.
Null (default)
locInfo.x.PRD
   Enter the leading direction of the street location.
     Null (default)

locInfo.x.RD
   Enter the name of road or street where the phone is located.
     Null (default)

locInfo.x.STS
   Enter the suffix of the name used in locInfo.x.RD. For example, street or avenue.
     Null (default)

locInfo.x.POD
   Enter the trailing street direction. For example, southwest.
     Null (default)

locInfo.x.HNO
   Enter the street address number of the phone’s location.
     Null (default)

locInfo.x.HNS
   Enter a suffix for the street address used in locInfo.x.HNS. For example, A or ½.
     Null (default)

locInfo.x.LOC
   Enter any additional information that identifies the location.
     Null (default)

locInfo.x.NAM
   Enter a proper name to associate with the location.
     Null (default)

locInfo.x.PC
   Enter the ZIP or postal code of the phone’s location.
     Null (default)

feature.E911.enabled
   0 (default) - Disable the E.911 feature.
   1 - Enable the E.911 feature.
The INVITE sent for emergency calls from the phone includes the geolocation header defined in RFC 6442 and PIDF presence element as specified in RFC3863 with a GEOPRIV location object specified in RFC4119 for in Open SIP environments.

This parameter is mutually exclusive of the Ribbon Communications E.911 feature and if this parameter and feature.genband.E911.enabled are enabled, this parameter takes precedence.

**feature.E911.HELD.requestType**

Any (default) - Send a request to the Location Information Server (LIS) to return either ‘Location by Reference’ or ‘Location by Value’. Note this is not the ‘Any’ value referred to in RFC 5985.

Civic - Send a request to the LIS to return a location by value in the form of a civic address for the device as defined in RFC 5985.

RefID - Send a request to the LIS to return a set of Location URIs for the device as defined in RFC 5985.

**voIpProt.SIP.header.priority.enable**

0 (default) – Do not include a priority header in the E.911 INVITE message.

1 - Include a priority header in the E.911 INVITE message.

**voIpProt.SIP.header.geolocation-routing.enable**

0 (default) – Do not include the geolocation-routing header in the E.911 INVITE message.

1 - Include the geolocation-routing header in the E.911 INVITE message.

**feature.E911.HELD.secondary.server**

Set the IP address or hostname of the secondary Location Information Server (LIS) address. For example, host.domain.com or https://xxx.xxx.xxx.xxx.

NULL (default)

0-255

Dotted-decimal IP address

Hostname

Fully-qualified domain name (FQDN)

**feature.E911.HELD.secondary.username**

Set a user name to authenticate to the secondary Location information Server (LIS).

NULL (default)

String

0-255

**feature.E911.HELD.secondary.password**

Set a password to authenticate to the secondary LIS.

NULL (default)
feature.E911.usagerule.retransmission

0 (default) - The recipient of this location object is not permitted to share the enclosed location information, or the object as a whole, with other parties.

1 - Distributing this location is permitted.

**MLPP for AS-SIP**

Multilevel Precedence and Preemption (MLPP) enables you to configure a precedence level for outgoing calls, which is implemented in accordance with the standards set by Assured Services for Session Initiation Protocol (AS-SIP).

Higher precedence calls preempt—end—active calls with a lower precedence level. When an active call is preempted, the phone plays a preemption tone and displays a preemption screen. The preemption screen display time can be configured in the configuration file. The default time for the preempted screen is 0 seconds for callee and 3 seconds for caller. If the default time for the preempted screen is 0 seconds, then preemption screen is displayed until you press the OK button. The preemption screen shows that the current call was preempted, and an OK button to acknowledge the preemption. The user can then answer the incoming higher-precedence call or reject the call. If the callee doesn't acknowledge the incoming call, the notification disappears and the current call ends.

If a lower-precedence call is on hold, and you receive a higher-precedence call, the preemption screen doesn't display, and the preemption tone doesn't play.

MLPP treats incoming calls with the same precedence level as the active call depending on the call state, as shown in the following table.

**MLPP Behavior**

<table>
<thead>
<tr>
<th>Current Call State</th>
<th>New call—same precedence: one active call</th>
<th>New call—same precedence: multiple active calls</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active Call</td>
<td>Rejected</td>
<td>If you accept the new call, it’s placed in the first slot. The active call is placed on hold and moved to the second slot. If all lines and call appearances are at capacity, new incoming call with the same precedence will get rejected.</td>
</tr>
<tr>
<td>Ringing State</td>
<td>Rejected</td>
<td>The new call displays in the top center corner and the current call is in the main screen.</td>
</tr>
</tbody>
</table>
## Call Controls

<table>
<thead>
<tr>
<th>Current Call State</th>
<th>New call—same precedence: one active call</th>
<th>New call—same precedence: multiple active calls</th>
</tr>
</thead>
<tbody>
<tr>
<td>Call on Hold</td>
<td>Rejected</td>
<td>If the user acknowledges the new call, the current call is moved to the second slot. The new call is placed in the first slot.</td>
</tr>
</tbody>
</table>

The caller’s phone displays the precedence of the outgoing call. Callee phones display call precedence on each phone line: 1 indicates the lowest precedence and 5 indicates the highest precedence.

Phone models vary in how they display precedence:

### Setting Call Precedence with a Digit Map

To set call precedence you can configure a digit map in a dial plan or you can set precedence directly on the server using a number prefix and a namespace. If you enable MLPP and don’t configure a precedence level to a number, the phone sends the call at the default precedence level.

**Example**

dialplan.digitmap=3xxxTP4|4xxxTP5

All digits matching the pattern 3xxxTP4 and 4xxxTP5 are sent with precedence 4 and 5 respectively. In this example:

- If the user dials 3434 the call is sent with precedence 4.
- If the user dials 3345 the call is sent with precedence 4.
- If the user dials 4666 the call is sent with precedence 5.

If the dialed number doesn’t match any number, then the default precedence is sent based on the value you set with voIpProt.SIP.assuredService.defaultPriority, default value 1.

### Preemption Behavior on Low Priority Calls

A 180 ringing response is sent to the far end only when a call appearance is allocated for the incoming precedence call.

The following table illustrates the preemption behavior of the low priority call's status.

#### Preemption Behavior on Low Priority Calls

<table>
<thead>
<tr>
<th>Low Priority Call's Status for Preemption</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connected</td>
<td>The call is terminated with a BYE request containing a preemption Reason header, and a local preemption tone is played for a configurable duration or until the user hangs up, whichever comes first.</td>
</tr>
<tr>
<td>Locally Held</td>
<td>The call may be terminated with a BYE request containing a preemption Reason header.</td>
</tr>
</tbody>
</table>
### Low Priority Call’s Status for Preemption

<table>
<thead>
<tr>
<th>Status</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alerting</td>
<td>A 486 Busy Here response is sent to the far end containing a preemption Reason header.</td>
</tr>
<tr>
<td>Dial Tone or Setup</td>
<td>When the final call appearance is in the dial tone or setup (digit collection) state (including consultation calls) and a precedence call arrives, no action is taken until the new outgoing call is of higher priority or is not is determined. If the call is of lower priority, then the call is not placed and a preemption tone is played for a configurable duration or until the user hangs up, whichever is less. If the call is of the same or higher priority, then the incoming call is terminated by sending a 486 Busy Here response to the far end containing a preemption Reason header.</td>
</tr>
<tr>
<td>Preceding</td>
<td>If the final call appearance is in the preceding (digit collection) state (including consultation calls) when a precedence call arrives, no action is taken until it can be determined whether the new outgoing call is of higher priority or not. If the call is determined to be of lower priority, then the call is not placed and a preemption tone should be played for a configurable duration or until the user hangs up, whichever is less. If the call is determined to be of the same or higher priority, then the incoming call is terminated by sending a 486 Busy Here response to the far end containing a preemption Reason header.</td>
</tr>
</tbody>
</table>

### MLPP with Shared Lines

MLPP interacts with the phone’s display of shared line call appearances and uses the following precedence rules:

- When a call is active, incoming higher-precedence call preempts the current active call.
- If a lower-precedence call is on hold, the preemption screen for the current call doesn’t display.

**Note:** Poly recommends you not to change the default value of parameter `callsPerLineKey`, which may result in improper functionality of MLPP feature for shared lines.

### MLPP with Call Transfer

MLPP phone behavior varies with the type of transfer:

- **Blind transfer** – Call precedence isn’t sent when you perform a blind transfer.
- **Consultative transfer** – Calls sent via consultative transfer are sent with the highest precedence level between the caller and the user transferring the call.
MLPP with Conference Calls
Conference calls occupy one call appearance and are treated as a single call. The precedence level of the conference is determined by the precedence of the highest-precedence participant.

New calls received during a conference call are handled according to their precedence level.

MLPPP with n-way Conference Calls
The phone displays the precedence level of each participant. The precedence level of the conference is determined by the precedence of the highest-precedence participant.

MLPP with AS-SIP Parameters
The following parameters configure MLPP with AS-SIP.

voIpProt.SIP.assuredService.defaultPriority
   Default priority assigned to an outgoing call.
   1 (default)
   1 to 10
   This value is overridden if priority is assigned from the dial plan for that number.

voIpProt.SIP.assuredService.enable
   0 (default) - Disables the AS-SIP feature.
   1 - Enables the AS-SIP feature

voIpProt.SIP.assuredService.namespace.custom.name
   The name for the custom namespace label.
   Null (default)
   String

voIpProt.SIP.assuredService.namespace.custom.priority.x
   The namespace precedence values, lowest to highest.
   Null (default)
   String

voIpProt.SIP.assuredService.precedenceThreshold
   The minimum call priority required for a call to be treated as a precedence call.
   2 (default)
   1 to 10

voIpProt.SIP.assuredService.preemptionAutoTerminationDelay.local
   Set the duration after a callee preemption event that a call appearance is automatically cleared.
voIpProt.SIP.assuredService.preemptionAutoTerminationDelay.remote
Set the duration after a caller preemption event that a call appearance is automatically cleared.
3 (default)
0-3600

voIpProt.SIP.assuredService.serverControlled
1 (default) - The precedence level of outgoing calls is set by the server or non-EI equipment.
0 - The precedence level is set by the phone and must not change if it is an outgoing call.

AS-SIP Namespace Parameter
Use the parameter below to configure the AS-SIP namespace scheme.

voIpProt.SIP.assuredService.namespace
The namespace scheme to use in SIP signaling.
UCRdsn (default)
dsn
drsn
UCRdrsn
custom
ets

International Dialing Prefix
Enter a plus (+) symbol before you dial an international phone numbers to identify to the switch that you are dialing an international phone number.

International Dialing Prefix Parameters
The following parameters configure the international dialing prefixes.

call.internationalDialing.enabled
This parameter applies to all numeric dial pads on the phone, including for example, the contact directory.
Changes you make to this parameter cause a restart or reboot.
1 (default) - Disable the key tap timer that converts a double tap of the asterisk "*" symbol to the "+" symbol to indicate an international call. By default, this parameter is enabled so that a quick double tap of "**" converts immediately to "+". To enter a double asterisk "***", tap "**" once and wait for the key tap timer to expire to enter a second "*".

0 - When you disable this parameter, you cannot dial "+" and you must enter the international exit code of the country you are calling from to make international calls.

Change causes system to restart or reboot.

call.internationalPrefix.key

The phone supports international call prefix (+) with both "0" and "*".

0 (default) - Set the international prefix with "*".
1 - Set the international prefix with "0".
Shared Lines

Topics:

- Shared Call Appearances
- Private Hold on Shared Lines
- Intercom Calls
- Push-to-Talk
- Group Paging
- SIP-B Automatic Call Distribution

This section shows you how to configure shared line features.

Shared Call Appearances

Shared call appearance enables calls to display simultaneously on multiple phones in a group.

All call states—active, inactive, on hold—are displayed on all phones of a group.

By default, the answering phone has sole access to the incoming call, which is called line seize. If the answering phone places the call on hold, that call becomes available for pickup to all phones in that group. You can enable other phones in the group to enter a conversation on one of the group phones. This is referred to as a barge in.

**Note:** Shared call appearances and bridged line appearances are similar signaling methods that enable more than one phone to share the same line or registration. The method you use varies with the SIP call server you are using.

Shared Call Appearances Parameters

This feature is dependent on support from a SIP call server. To enable shared call appearances on your phone, you must obtain a shared line address from your SIP service provider.

A shared line is an address of record managed by a call server. The server allows multiple endpoints to register locations against the address of record.

Poly devices support Shared Call Appearance (SCA) using the SUBSCRIBE-NOTIFY method specified in RFC 6665. The events used are:

- Call-info for call appearance state notification
- Line-seize for the phone to ask to seize the line

Use the parameters in the following list to configure options for this feature.

**reg.x.address**

The user part (for example, 1002) or the user and the host part (for example, 1002@polycom.com) of the registration SIP URI.

Null (default)
string address

reg.x.type
private (default) - Use standard call signaling.
shared - Use augment call signaling with call state subscriptions and notifications and use access control for outgoing calls.

call.shared.reject
For shared line calls on the BroadWorks server.
0 - The phone displays a Reject soft key to reject an incoming call to a shared line.
1 - The Reject soft key does not display.

call.shared.exposeAutoHolds
0 (default) - No re-INVITE is sent to the server when setting up a conference on a shared line.
1 - A re-INVITE is sent to the server when setting up a conference on a shared line.

call.shared.preferCallInfoCID
0 (default) - The Caller-ID information received in the 200 OK status code is not ignored if the NOTIFY message received with caller information includes display information.
1 - The Caller-ID information received in the 200 OK status code is ignored if the NOTIFY message received with caller information includes display information.

call.shared.remoteActiveHoldAsActive
1 (default) - Shared remote active/hold calls are treated as a active call on the phone.
0 - Shared remote active/hold calls are not treated as a active call on the phone.

call.shared.seizeFailReorder
1 (default) - Play a re-order tone locally on shared line seize failure.
0 - Do not play a re-order tone locally on shared line seize failure.
Change causes system to restart or reboot.

voIPProt.SIP.specialEvent.lineSeize.nonStandard
Controls the response for a line-seize event SUBSCRIBE.
1 (default) - This speeds up the processing of the response for line-seize event.
0 - This will process the response for the line seize event normally
Change causes system to restart or reboot.

reg.x.ringType
The ringer to be used for calls received by this registration. The default is the first non-silent ringer.
If you use the configuration parameters ringer13 and ringer14 on a single registered line, the phone plays SystemRing.wav.

default (default)

ringer1 to ringer24

**reg.x.line.y.label**

Configure a unique line label for a shared line that has multiple line key appearances. This parameter takes effect when up.cfgUniqueLineLabel=1. If reg.x.linekeys=1, this parameter does not have any effect.

x = the registration index number starting from 1.

y = the line index from 1 to the value set by reg.x.linekeys. Specifying a string sets the label used for the line key registration on phones with multiple line keys.

If no parameter value is set for reg.x.line.y.label, the phone automatically numbers multiple lines by prepending "<y>_" where <y> is the line index from 1 to the value set by reg.x.linekeys.

**reg.x.header.pearlymedia.support**

0 (Default) - The p-early-media header is not supported on the specified line registration.

1 - The p-early-media header is supported by the specified line registration.

**reg.x.insertOBPAddressInRoute**

1 (Default) - The outbound proxy address is added as the topmost route header.

0 - The outbound proxy address is not added to the route header.

**reg.x.path**

0 (Default) - The path extension header field in the Register request message is not supported for the specific line registration.

1 - The phone supports and provides the path extension header field in the Register request message for the specific line registration.

**reg.x.regevent**

0 (default) - The phone is not subscribed to registration state change notifications for the specific phone line.

1 - The phone is subscribed to registration state change notifications for the specific phone line.

This parameter overrides the global parameter voIpProt.SIP.regevent.

**reg.x.rejectNDUBInvite**

Specify whether or not the phone accepts a call for a particular registration in case of a Network Determined User Busy (NDUB) event advertised by the SIP server.

0 (Default) - If an NDUB event occurs, the phone does not reject the call.

1 - If an NDUB event occurs, the phone rejects the call with a 603 Decline response code.
**reg.x.server.y.specialInterop**

Specify the server-specific feature set for the line registration.

- Standard (Default)
- Standard
- GENBAND
- ALU-CTS
- ocs2007r2
- lcs2005

**reg.x.gruu**

- 1 - The phone sends sip.instance in the REGISTER request.
- 0 (default) - The phone does not send sip.instance in the REGISTER request.

**reg.x.serverFeatureControl.securityClassification**

- 0 (default) - The visual security classification feature for a specific phone line is disabled.
- 1 - The visual security classification feature for a specific phone line is enabled.

**reg.x.acd-login-logout reg.x.acd-agent-available**

- 0 (default) - The ACD feature is disabled for registration.
- 1 - If both ACD login/logout and agent available are set to 1 for registration x, the ACD feature is enabled for that registration.

**reg.x.auth.domain**

The domain of the authorization server that is used to check the user names and passwords.

- Null (default) string

**reg.x.auth.optimizedInFailover**

The destination of the first new SIP request when failover occurs.

- 0 (default) - The SIP request is sent to the server with the highest priority in the server list.
- 1 - The SIP request is sent to the server which sent the proxy authentication request.

**reg.x.auth.password**

The password to be used for authentication challenges for this registration.

- Null (default)
- string - It overrides the password entered into the Authentication submenu on the Settings menu of the phone.

**reg.x.auth.userId**

User ID to be used for authentication challenges for this registration.
Null (default)

string - If the User ID is non-Null, it overrides the user parameter entered into the Authentication submenu on the Settings menu of the phone.

\texttt{reg.x.auth.useLoginCredentials}

0 - (default) The Login credentials are not used for authentication to the server on registration \textit{x}.
1 - The login credentials are used for authentication to the server.

\texttt{reg.x.broadsoft.userId}

Enter the BroadSoft user ID to authenticate with the BroadSoft XSP service interface.
Null (default)
string

\texttt{reg.x.broadsoft.useXspCredentials}

If this parameter is disabled, the phones use standard SIP credentials to authenticate.
1 (default) - Use this value, if phone lines are registered with a server running BroadWorks R19 or earlier.
0 - Set to 0, if phone lines are registered with a server running BroadWorks R19 SP1 or later.

\texttt{reg.x.broadsoft.xsp.password}

Enter the password associated with the BroadSoft user account for the line. Required only when \texttt{reg.x.broadsoft.useXspCredentials=1}.
Null (default)
string

\texttt{reg.x.displayName}

The display name used in SIP signaling and/or the H.323 alias used as the default caller ID.
Null (default)
UTF-8 encoded string

\texttt{reg.x.enablePvtHoldSoftKey}

This parameter applies only to shared lines.
0 (default) - To disable user on a shared line to hold calls privately.
1 - To enable users on a shared line to hold calls privately.

\texttt{reg.x.filterReflectedBlaDialogs}

1 (default) - bridged line appearance NOTIFY messages are ignored.
0 - bridged line appearance NOTIFY messages is not ignored
**reg.x.fwd.busy.contact**

The forward-to contact for calls forwarded due to busy status.

- Null (default) - The contact specified by `divert.x.contact` is used.
- String - The contact specified by `divert.x.contact` is not used

**reg.x.fwd.busy.status**

- 0 (default) - Incoming calls that receive a busy signal is not forwarded
- 1 - Busy calls are forwarded to the contact specified by `reg.x.fwd.busy.contact`.

**reg.x.fwd.noanswer.contact**

- Null (default) - The forward-to contact specified by `divert.x.contact` is used.
- String - The forward to contact used for calls forwarded due to no answer.

**reg.x.fwd.noanswer.ringCount**

The number of seconds the phone should ring for before the call is forwarded because of no answer. The maximum value accepted by some call servers is 20.

- 0 - (default)
- 1 to 65535

**reg.x.fwd.noanswer.status**

- 0 (default) - The calls are not forwarded if there is no answer.
- 1 - The calls are forwarded to the contact specified by `reg.x.noanswer.contact` after ringing for the length of time specified by `reg.x.fwd.noanswer.ringCount`.

**reg.x.gruu**

Specify if the phone sends sip.instance in the REGISTER request.

- 0 (default)
- 1

**reg.x.label**

The text label that displays next to the line key for registration x.

The maximum number of characters for this parameter value is 256; however, the maximum number of characters that a phone can display on its user interface varies by phone model and by the width of the characters you use. Parameter values that exceed the phone’s maximum display length are truncated by ellipses (...). The rules for parameter `up.cfgLabelElide` determine how the label is truncated.

Null (default) - the label is determined as follows:

- If `reg.1.seteluriAsLineLabel=1`, then the tel URI(phone number/address) displays as the label.
- If \texttt{reg.1.useTelURIAsLineLabel=0}, then the value for \texttt{reg.x.displayName}, if available, displays as the label. If \texttt{reg.x.displayName} is unavailable, the user part of \texttt{reg.x.address} is used.

\texttt{UTF-8} encoded string

\texttt{reg.x.lineAddress}

The line extension for a shared line. This parameter applies to private lines and BroadSoft call park and retrieve. If there is no extension provided for this parameter, the call park notification is ignored for the shared line.

Null (default)

String

\texttt{reg.x.lineKeys}

Specify the number of line keys to use for a single registration. The maximum number of line keys you can use per registration depends on your phone model.

1 (default)

1 to max

\texttt{reg.x.lisdisclaimer}

This parameter sets the value of the location policy disclaimer. For example, the disclaimer may be “Warning: If you do not provide a location, emergency services may be delayed in reaching your location should you need to call for help.”

Null (default)

string, 0 to 256 characters

\texttt{reg.x.musicOnHold.uri}

A URI that provides the media stream to play for the remote party on hold.

Null (default) - This parameter does not overrides \texttt{voIpProt.SIP.musicOnHold.uri}.

A SIP URI - This parameter overrides \texttt{voIpProt.SIP.musicOnHold.uri}.

\texttt{reg.x.offerFullCodecListUponResume}

1 (default) - The phone sends full audio and video capabilities after resuming a held call irrespective of the audio and video capabilities negotiated at the initial call answer.

0 - The phone does not send full audio and video capabilities after resuming a held call.

\texttt{reg.x.outboundProxy.address}

The IP address or hostname of the SIP server to which the phone sends all requests.

Null (default)

IP address or hostname
**reg.x.outboundProxy.failOver.failBack.mode**

The mode for failover failback (overrides reg.x.server.y.failOver.failBack.mode).

- **duration** - (default) The phone tries the primary server again after the time specified by reg.x.outboundProxy.failOver.failBack.timeout expires.
- **newRequests** - All new requests are forwarded first to the primary server regardless of the last used server.
- **DNSTTL** - The phone tries the primary server again after a timeout equal to the DNS TTL configured for the server that the phone is registered to.

**reg.x.outboundProxy.failOver.failBack.timeout**

- 3600 (default) - The time to wait (in seconds) before failback occurs (overrides reg.x.server.y.failOver.failBack.timeout).
- 0, 60 to 65535 - The phone does not fail back until a failover event occurs with the current server.

**reg.x.outboundProxy.failOver.failRegistrationOn**

- 1 (default) - The reRegisterOn parameter is enabled, the phone silently invalidates an existing registration.
- 0 - The reRegisterOn parameter is enabled, existing registrations remain active.

**reg.x.outboundProxy.failOver.onlySignalWithRegistered**

- 1 (default) - The reRegisterOn and failRegistrationOn parameters are enabled, no signaling is accepted from or sent to a server that has failed until failback is attempted or failover occurs.
- 0 - The reRegisterOn and failRegistrationOn parameters are enabled, signaling is accepted from and sent to a server that has failed.

**reg.x.outboundProxy.failOver.reRegisterOn**

This parameter overrides reg.x.server.y.failOver.reRegisterOn.

- 0 (default) - The phone won't attempt to register with the secondary server.
- 1 - The phone attempts to register with (or via, for the outbound proxy scenario), the secondary server.

**reg.x.outboundProxy.port**

The port of the SIP server to which the phone sends all requests.

- 0 - (default)
- 1 to 65535

**reg.x.outboundProxy.transport**

The transport method the phone uses to communicate with the SIP server.

- DNSnaptr (default)
- DNSnaptr, TCPpreferred, UDPOnly, TLS, TCPOnly
reg.x.proxyRequire
Null (default) - No Proxy-Require is sent.
string - Needs to be entered in the Proxy-Require header.

reg.x.ringType
The ringer to be used for calls received by this registration.
ringer2 (default) - Is the first non-silent ringer.
ringer1 to ringer24 - To play ringer on a single registered line.

reg.x.serverFeatureControl.callRecording
1 (default) - BroadSoft BroadWorks v20 call recording feature for individual phone lines is enabled.
0 - BroadSoft BroadWorks v20 call recording feature for individual phone lines is disabled.

reg.x.serverFeatureControl.cf
0 (default) - The server-based call forwarding is disabled.
1 - server based call forwarding is enabled.

Note: This parameter overrides voIpProt.SIP.serverFeatureControl.cf.
Change causes system to restart or reboot.

reg.x.serverFeatureControl.dnd
0 (default) - server-based do-not-disturb (DND) is disabled.
1 - server-based DND is enabled and the call server has control of DND.

Note: This parameter overrides voIpProt.SIP.serverFeatureControl.dnd.
Change causes system to restart or reboot.

reg.x.serverFeatureControl.localProcessing.cf
0 (default) - If reg.x.serverFeatureControl.cf is set to 1 the phone does not perform local Call Forward behavior.
1 - The phone performs local Call Forward behavior on all calls received.

Note: This parameter overrides voIpProt.SIP.serverFeatureControl.localProcessing.cf.

reg.x.serverFeatureControl.localProcessing.dnd
0 (default) - If reg.x.serverFeatureControl.dnd is set to 1, the phone does not perform local DND call behavior.
1 - The phone performs local DND call behavior on all calls received.
Note: This parameter overrides
voIPProt.SIP.serverFeatureControl.localProcessing.dnd.

reg.x.serverFeatureControl.securityClassification
0 (default) - The visual security classification feature for a specific phone line is disabled.
1 - The visual security classification feature for a specific phone line is enabled.

reg.x.serverFeatureControl.signalingMethod
Controls the method used to perform call forwarding requests to the server.
serviceMsForwardContact (default)
string

reg.x.srtp.enable
1 (default) - The registration accepts SRTP offers.
0 - The registration always declines SRTP offers.
Change causes system to restart or reboot.

reg.x.srtp.offer
This parameter applies to the registration initiating (offering) a phone call.
0 (default) - No secure media stream is included in SDP of a SIP INVITE.
1 - The registration includes a secure media stream description along with the usual non-secure
media description in the SDP of a SIP INVITE.
Change causes system to restart or reboot.

reg.x.srtp.require
0 (default) - Secure media streams are not required.
1 - The registration is only allowed to use secure media streams.
Change causes system to restart or reboot.

reg.x.srtp.simplifiedBestEffort
1 (default) - Negotiation of SRTP compliant with Microsoft Session Description Protocol Version
2.0 Extensions is supported.
0 - No SRTP is supported.

Note: This parameter overrides sec.srtp.simplifiedBestEffort.

reg.x.strictLineSeize
0 (default) - Dial prompt is provided immediately without waiting for a successful OK from the
call server.
1 - The phone is forced to wait for 200 OK on registration x when receiving a TRYING notify.

**Note:** This parameter overrides `voIpProt.SIP.strictLineSeize` for registration x.

**reg.x.tcpFastFailover**

0 (default) - A full 32 second RFC compliant timeout is used.

1 - Failover occurs based on the values of `reg.x.server.y.retryMaxCount` and `voIpProt.server.x.retryTimeOut`.

**reg.x.thirdPartyName**

Null (default) - In all other cases.

String address - This field must match the `reg.x.address` value of the registration which makes up the part of a bridged line appearance (BLA).

**reg.x.useCompleteUriForRetrieve**

1 (default) - The target URI in BLF signaling uses the complete address as provided in the XML dialog document.

0 - Only the user portion of the XML dialog document is used and the current registrar's domain is appended to create the full target URI.

**Note:** This parameter overrides `voipPort.SIP.useCompleteUriForRetrieve`.

**reg.x.server.y.address**

If this parameter is set, it takes precedence even if the DHCP server is available.

Null (default) - SIP server does not accept registrations.

IP address or hostname - SIP server that accepts registrations. If not Null, all of the parameters in this list override the parameters specified in `voIpProt.server.*`.

**reg.x.server.y.expires**

The phone's requested registration period in seconds.

The period negotiated with the server may be different. The phone attempts to re-register at the beginning of the overlap period.

3600 - (default)

Positive integer, minimum 10

**reg.x.server.y.expires.lineSeize**

Requested line-seize subscription period.

30 - (default)

0 to 65535
**reg.x.server.y.expires.overlap**

The number of seconds before the expiration time returned by server x at which the phone should try to re-register.

The phone tries to re-register at half the expiration time returned by the server if the server value is less than the configured overlap value.

60 (default)

5 to 65535

**reg.x.server.y.failOver.failBack.mode**

duration (default) - The phone tries the primary server again after the time specified by `reg.x.server.y.failOver.failBack.timeout`.

newRequests - All new requests are forwarded first to the primary server regardless of the last used server.

DNSTTL - The phone tries the primary server again after a timeout equal to the DNS TTL configured for the server that the phone is registered to.

registration - The phone tries the primary server again when the registration renewal signaling begins.

This parameter overrides `voIpProt.server.x.failOver.failBack.mode`

**reg.x.server.y.failOver.failBack.timeout**

3600 (default) - The time to wait (in seconds) before failback occurs.

0 - The phone does not fail back until a failover event occurs with the current server.

60 to 65535 - If set to Duration, the phone waits this long after connecting to the current working server before selecting the primary server again.

**reg.x.server.y.failOver.failRegistrationOn**

1 (default) - The reRegisterOn parameter is enabled, the phone silently invalidates an existing registration (if it exists), at the point of failing over.

0 - The reRegisterOn parameter is disabled, existing registrations remain active.

**reg.x.server.y.failOver.onlySignalWithRegistered**

1 (default) - Set to this value and reRegisterOn and failRegistrationOn parameters are enabled, no signaling is accepted from or sent to a server that has failed until failback is attempted or failover occurs. If the phone attempts to send signaling associated with an existing call via an unregistered server (for example, to resume or hold a call), the call ends. No SIP messages are sent to the unregistered server.

0 - Set to this value and reRegisterOn and failRegistrationOn parameters are enabled, signaling is accepted from and sent to a server that has failed (even though failback hasn't been attempted or failover hasn't occurred).

**reg.x.server.y.failOver.reRegisterOn**

0 (default) - The phone does not attempt to register with the secondary server, since the phone assumes that the primary and secondary servers share registration information.
1 - The phone attempts to register with (or via, for the outbound proxy scenario), the secondary server. If the registration succeeds (a 200 OK response with valid expires), signaling proceeds with the secondary server.

This parameter overrides voIpProt.server.x.failOver.reRegisterOn.

**reg.x.server.y.port**

Null (default) - The port of the SIP server does not specifies registrations.

0 - The port used depends on reg.x.server.y.transport.

1 to 65535 - The port of the SIP server that specifies registrations.

**reg.x.server.y.register**

1 (default) - Calls can not be routed to an outbound proxy without registration.

0 - Calls can be routed to an outbound proxy without registration.

See voIpProt.server.x.register for more information, see SIP Server Fallback Enhancements on Polycom Phones - Technical Bulletin 5844 on Polycom Engineering Advisories and Technical Notifications.

**reg.x.server.y.registerRetry.baseTimeOut**

For registered line x, set y to the maximum time period the phone waits before trying to re-register with the server. Used in conjunction with

reg.x.server.y.registerRetry.maxTimeOut to determine how long to wait.

60 (default)

10 - 120 seconds

**reg.x.server.y.registerRetry.maxTimeout**

For registered line x, set y to the maximum time period the phone waits before trying to re-register with the server. Use in conjunction with r
gex.x.server.y.registerRetry.baseTimeOut to determine how long to wait. The algorithm is defined in RFC 5626.

180 - (default)

60 - 1800 seconds

**reg.x.server.y.retryMaxCount**

The number of retries attempted before moving to the next available server.

3 - (default)

0 to 20 - 3 is used when the value is set to 0.

**reg.x.server.y.retryTimeOut**

0 (default) - Use standard RFC 3261 signaling retry behavior.

0 to 65535 - The amount of time (in milliseconds) to wait between retries.
**reg.x.server.y.subscribe.expires**

The phone's requested subscription period in seconds after which the phone attempts to resubscribe at the beginning of the overlap period.

- 3600 seconds - (default)
- 10 - 2147483647 (seconds)

You can use this parameter in conjunction with reg.x.server.y.subscribe.expires.overlap.

**reg.x.server.y.subscribe.expires.overlap**

The number of seconds before the expiration time returned by server x after which the phone attempts to resubscribe. If the server value is less than the configured overlap value, the phone tries to resubscribe at half the expiration time returned by the server.

- 60 seconds (default)
- 5 - 65535 seconds

**reg.x.server.y.transport**

The transport method the phone uses to communicate with the SIP server.

- DNSnaptr (default) - If reg.x.server.y.address is a hostname and reg.x.server.y.port is 0 or Null, do NAPTR then SRV look-ups to try to discover the transport, ports and servers, as per RFC 3263. If reg.x.server.y.address is an IP address, or a port is given, then UDP is used.
- TCPpreferred - TCP is the preferred transport; UDP is used if TCP fails.
- UDPOnly - Only UDP is used.
- TLS - If TLS fails, transport fails. Leave port field empty (defaults to 5061) or set to 5061.
- TCPOnly - Only TCP is used.

**reg.x.server.y.useOutboundProxy**

- 1 (default) - Enables to use the outbound proxy specified in reg.x.outboundProxy.address for server x.
- 0 - Disable to use the outbound proxy specified in reg.x.outboundProxy.address for server x.

**divert.x.sharedDisabled**

- 1 (default) - Disables call diversion features on shared lines.
- 0 - Enables call diversion features on shared lines.

Change causes system to restart or reboot.

**call.shared.distinctiveLedOnHold**

- 0 (default) - The LED blinks red for both remotely held calls and locally held calls.
- 1 - The LED blinks as red and green for local hold calls, and blinks only red for remotely held calls.
Private Hold on Shared Lines

Enable the private hold feature to enable users to hold calls without notifying other phones registered with the shared line.

When you enable the feature, users can hold a call, transfer a call, or initiate a conference call and the shared line displays as busy to others sharing the line.

Private Hold on Shared Lines Parameters

You can configure private hold only using configuration files; you cannot configure the feature on the Web Configuration Utility or from the local phone interface.

Use the parameters in the following list to configure this feature.

`call.shared.exposeAutoHolds`

   Enable to send a re-INVITE to the server when setting up a conference on a shared line.
   0 (default) - Disabled
   1 - Enabled
   Change causes system to restart or reboot.

`reg.x.enablePvtHoldSoftKey`

   Enable to allow users on a shared line to hold calls privately.
   0 (default) - Disabled
   1 - Enabled

   **Note:** This parameter applies only to shared lines.

Intercom Calls

The Intercom feature enables users to place an intercom call that is answered automatically on the dialed contact's phone.

This is a server-independent feature provided the server does not alter the Alert-Info header sent in the INVITE.

Creating a Custom Intercom Soft Key

By default, an Intercom soft key displays on the phone, but you have the option to provide users the ability to initiate intercom calls directly to a specified contact using enhanced feature keys (EFKs).

You do not need to disable the default Intercom soft key to create a custom soft key.

For example, you can create an intercom action string for a custom soft key in one of the following ways:

- `$FIntercom$`
This is an F type macro that behaves as a custom Intercom soft key. Pressing the soft key opens the Intercom dial prompt users can use to place an Intercom call by entering the destination's digits and using a speed dial or BLF button.

- `<number>$Tintercom$`

This is a T type macro that enables you to specify a Direct intercom button that always calls the number you specify in `<number>`. No other input is necessary.

**Intercom Calls Parameters**

Use the parameters in the table to configure the behavior of the calling and answering phone.

**Intercom Parameters**

<table>
<thead>
<tr>
<th>Template</th>
<th>Parameter</th>
<th>Permitted Values</th>
<th>Change Causes Restart or Reboot</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>features.cfg</code></td>
<td><code>feature.intercom.enable</code></td>
<td>0 (default) - Disable the Intercom feature. 1 - Enable the Intercom feature.</td>
<td>No</td>
</tr>
<tr>
<td><code>features.cfg</code></td>
<td><code>homeScreen.intercom.enable</code></td>
<td>1 (default) - Enable the Intercom icon on the phone Home screen. 0 - Disable the Intercom icon on the phone Home screen.</td>
<td>No</td>
</tr>
<tr>
<td><code>sip-interop.cfg</code></td>
<td><code>voIpProt.SIP.intercom.m.alertInfo</code></td>
<td>The string you want to use in the Alert-Info header. You can use the following characters: '@', ',', ';', ':' . If you use any other characters, NULL, or empty spaces, the call is sent as normal without the Alert-Info header. Intercom (default) Alpha - Numeric string</td>
<td>No</td>
</tr>
</tbody>
</table>

**Push-to-Talk**

The push-to-talk (PTT) is a collaborative tool that enables users to exchange broadcasts to users subscribed to any of the 25 PTT channels, much like a walkie-talkie.

Users can transmit pages and PTT broadcasts using their handset, headset, or speakerphone. PTT broadcasts can be received on the speakerphone, handset, and headset.

PTT mode is intended primarily for Wi-Fi phones. In PTT mode, the phone behaves like a walkie-talkie. Users can broadcast audio to a PTT channel and recipients subscribed to that channel can respond to messages.

You can specify the same IP multicast address in the parameter `ptt.address` for both PTT and group paging. Use the parameters in the following parameter list to configure this feature.
**Push-to-Talk Parameters**

Administrators must enable group paging and PTT before users can subscribe to a PTT channel.

PTT works in conjunction with group paging, and you can enable PTT or group paging, or enable both to operate simultaneously.

- **ptt.pttMode.enable**
  
  Enable or disabled push-to-talk.
  
  0 (default) - Disabled
  
  1 - Enabled

- **ptt.address**
  
  The multicast IP address to send page audio to and receive page audio from.
  
  224.0.1.116 (default)
  
  Multicast IP address.

- **ptt.allowOffHookPages**
  
  Enable to allow PTT messages to play on the phone while it is in an active call.
  
  0 (default) - Disabled. The user must accept incoming PTT messages to play out.
  
  1 - Enabled

- **ptt.callWaiting.enable**
  
  Enable to allow call waiting when incoming PTT calls come through on active audio channels.
  
  0 (default) - Disabled
  
  1 - Enabled

- **ptt.channel.x.allowReceive**
  
  Enable channel x to receive PTT calls.
  
  1 (default) - Enabled
  
  0 - Disabled

- **ptt.channel.x.allowTransmit**
  
  Enable outgoing PTT calls on channel x.
  
  1 (default) - Enabled
  
  0 - Disabled

---

**Note:** The push-to-talk and group paging features use an IP multicast address. If you want to change the default IP multicast address, ensure that the new address does not already have an official purpose as specified in the [IPv4 Multicast Address Space Registry](https://www.iana.org/assignments/multicast-addresses/multicast-addresses.xhtml).
**ptt.channel.x.available**
1 (default) - Channel x is available.
0 - Channel x is not available.

**ptt.channel.x.label**
Specify a label for channel x.
Null (default)
string

**ptt.channel.x.subscribed**
0 (default) - The PTT is not subscribed for channel x.
1 - 25 - The PTT is subscribed for channel x.

**ptt.codec**
Specify codec to use for PTT.
G.722 (default)
G.711Mu
G.726QI
G.722

**ptt.defaultChannel**
Specify the default channel number used for PTT transmissions.
1 (default)
1 - 25

**ptt.emergencyChannel**
Specify the channel to use for emergency PTT transmissions.
25 (default) 1 - 25

**ptt.emergencyChannel.volume**
Set the emergency page audio volume relative to the maximum speakerphone volume of the phone. Positive values are louder than the maximum and negative values are quieter. The gain to use for emergency page/PTT is the maximum termination gain plus this parameter. Note: To enter a negative number, press the * key first.
-10 (default)
-57 - 0

**ptt.port**
Specify the port values to send and receive audio.
5001 (default)
**ptt.displayName**

This display name is shown in the caller ID field of outgoing group pages. If Null, the value from reg.1.displayName is used.

NULL (default)
up to 64 octet UTF-8 string

**ptt.payloadSize**

Specify the payload size for PTT transmissions.

20 (default)
10
30
40
50
60
70
80

**ptt.priorityChannel**

Specify the channel number to use for priority PTT transmissions.

24 (default)
1 - 25

**ptt.volume**

Controls the volume level for pages without changing the volume level for incoming calls.

-20 (default)
-57 to 0

---

**Group Paging**

Group Paging enables users to make pages—one-way audio announcements—to users subscribed to a page group.

Group paging users can send announcements to recipients subscribed to any of the 25 paging groups. Any announcements sent to the paging group play through the phone’s speakerphone.

Administrators must enable paging before users can subscribe to a page group. You can specify the same IP multicast address in the parameter ptt.address for both PTT and paging mode.
**Note:** The push-to-talk and group paging features use an IP multicast address. If you want to change the default IP multicast address, ensure that the new address does not already have an official purpose as specified in the [IPv4 Multicast Address Space Registry](https://www.iana.org/assignments/address-registry-multicast-address-space).

### Group Paging Parameters

Administrators must enable paging and PTT before users can subscribe to a page group. Use the parameters in the following list to configure this feature.

#### ptt.address

The multicast IP address to send page audio to and receive page audio from.

- 224.0.1.116 (default)
- Multicast IP address.

#### ptt.pageMode.allowOffHookPages

Enable to play group pages on handsets while they are on active calls.

- 0 (default) - Disabled. Priority and Emergency pages still play while handsets are on active calls.
- 1 - Enabled.

#### ptt.pageMode.defaultGroup

The paging group used to transmit an outgoing page if the user does not explicitly specify a group.

- 1 (default)

1 to 25

#### ptt.pageMode.transmit.timeout.continuation

The time (in seconds) to add to the initial timeout (ptt.pageMode.transmit.timeout.initial) for terminating page announcements. If this value is non-zero, **Extend** displays on the phone. Pressing **Extend** continues the initial timeout for the time specified by this parameter. If 0, announcements cannot be extended.

- 60 (default)

0 to 65535

#### ptt.pageMode.transmit.timeout.initial

The number of seconds to wait before automatically terminating an outgoing page announcement.

- 0 (default) - The page announcements do not automatically terminate.

0 to 65535 - The page announcements automatically terminate.

#### ptt.pageMode.priorityGroup

The paging group to use for priority pages.
24 (default)
1 to 25

**ptt.pageMode.payloadSize**

The page mode audio payload size.
20 (default)
10, 20, ..., 80 milliseconds

**ptt.pageMode.emergencyGroup**

The paging group used for emergency pages.
25 (default)
1 to 25

**ptt.pageMode.codec**

The audio codec to use for outgoing group pages. Incoming pages are decoded according to the codec specified in the incoming message.
G.722 (default)
G.711Mu, G.726QI, or G.722

**ptt.pageMode.displayName**

This display name is shown in the caller ID field of outgoing group pages. If Null, the value from `reg.1.displayName` is used.
NULL (default)
up to 64 octet UTF-8 string

**ptt.pageMode.enable**

Enable or disable group paging.
0 (default) - Disabled
1 - Enabled

**ptt.pageMode.group.x.available**

Enable to make the group (x) available to the user.
1 (default) - Enabled
0 - Disabled

**ptt.pageMode.group.x.allowReceive**

Enable to allow the phone to receive pages from the group (x).
1 (default) - Enabled
0 - Disabled
**ptt.pageMode.group.x.allowTransmit**

Enable to allow outgoing announcements to the group.

1 (default) - Enabled

0 - Disabled

**ptt.pageMode.group.x.label**

The label to identify the group

ch24: Priority, ch25: Emergency, others:Null ch1, 24, 25: 1, others: 0 (default)

string

**ptt.pageMode.group.x.subscribed**

Subscribe the phone to the group.

A page mode group x, where x= 1 to 25. The label is the name used to identify the group during pages.

If available is disabled (0), the user cannot access the group or subscribe and the other page mode group parameters is ignored. If enabled, the user can access the group and choose to subscribe.

If allowTransmit is disabled (0), the user cannot send outgoing pages to the group. If enabled, the user may send outgoing pages.

1 (default) - If enabled, the phone subscribes to the group.

0 - If disabled, the phone does not subscribe to the group.

**SIP-B Automatic Call Distribution**

SIP-B Automatic Call Distribution enables you to use phones in a call center agent/supervisor role on a supported call server.

This feature supports ACD agent availability, which depends on support from a SIP server.

You can view or hide the menu items on the Automatic Call Distribution (ACD) menus. You can configure the phone to hide or display the ACD soft keys such as ASignIn or ASignOut, and Available.

**SIP-B Automatic Call Distribution Parameters**

Use the parameters in the following list to configure this feature.

**feature.acdLoginLogout.enabled**

Enable or disable the ACD login/logout feature.

0 (default) - Disabled

1 - Enabled

Change causes system to restart or reboot.
reg.x.acd-login-logout  reg.x.acd-agent-available

0 (default) - The ACD feature is disabled for registration.
1 - If both ACD login/logout and agent available are set to 1 for registration x, the ACD feature is enabled for that registration.

voIpProt.SIP.acd.signalingMethod

0 (default) - The 'SIP-B' signaling is supported. (This is the older ACD functionality.)
1 - The feature synchronization signaling is supported. (This is the new ACD functionality.)

Change causes system to restart or reboot.

acd.simplifiedAgentStateControl

0 (default) - Displays menu items.
1 - Hides ASignIN and associated soft keys. Also hides menu items under Menu > Settings > Feature > ACD.
User Profiles

Topics:

- User Profile Parameters
- Remotely Logging Out Users
- User Profile Authentication

When you set up user profiles, you enable users to access their personal phone settings, including their contact directory, speed dials, and other phone settings from any phone on the network.

This feature is useful for remote and mobile workers who don’t have a dedicated work space and conduct their business in more than one location. This feature is also useful if an office has a common conference phone from which multiple users need to access their personal settings.

---

**Note:** You can configure all company phones so that anyone can call authorized and emergency numbers when not logged in to a phone. For more information, see dialplan.routing.emergency.outboundIdentity.

---

If you set up the user profile feature, users can do the following:

- Log in to a phone to access their personal phone settings using their user ID and password.
- Place a call to an authorized number from a phone that is logged out.
- Change their user password.
- Log out of a phone after they finish using it.

If a user changes any settings while logged in to a phone, the settings save and display the next time the user logs in to another phone. When a user logs out, the corresponding user options are cleared from the device until the user profile related configuration is enabled on the phone again.

User Profile Parameters

Before you configure user profiles, you must complete the following:

- Create a phone configuration file, or update an existing file, to enable the feature's settings.
- Create a user configuration file in the format <user>.cfg to specify the user's password, registration, and other user-specific settings that you want to define.

**Important:** You can reset a user's password by removing the password parameter from the override file. This causes the phone to use the default password in the <user>.cfg file.

When you set up the user profile feature, you can set the following conditions:

- If users are required to always log in to use a phone and access their personal settings.
- If users are required to log in and have the option to use the phone as is without access to their personal settings.
- If users are automatically logged out of the phone when the phone restarts or reboots.
- If users remain logged in to the phone when the phone restarts or reboots.
User Profiles

Use the parameters in the following list to enable users to access their personal phone settings from any phone in an organization.

**prov.login.automaticLogout**
Specify the amount of time before a non-default user is logged out.
- 0 minutes (default)
- 0 to 46000 minutes

**prov.login.defaultOnly**
- 0 (default) - The phone cannot have users other than the default user.
- 1 - The phone can have users other than the default user.

**prov.login.defaultPassword**
Specify the default password for the default user.
- NULL (default)

**prov.login.defaultUser**
Specify the name of the default user. If a value is present, the user is automatically logged in when the phone boots up and after another user logs out.
- NULL (default)

**prov.login.enabled**
- 0 (default) - The user profile is disabled.
- 1 - The user profile feature is enabled.

**prov.login.localPassword.hashed**
- 0 (default) - The user's local password is formatted and validated as clear text.
- 1 - The user's local password is created and validated as a hashed value.

**prov.login.localPassword**
Specify the password used to validate the user login. The password is stored either as plain text or as an encrypted SHA1 hash.
- 123 (default)

**prov.login.persistent**
- 0 (default) - Users are logged out if the handset reboots.
- 1 - Users remain logged in when the phone reboots.

**prov.login.required**
Set whether the phone requires the user to log in to the phone to use it.
0 (default) - Login not required.
1 - Login is required.

**prov.login.useProvAuth**

0 (default) - The phone does not use server authentication.
1 - The phones use server authentication and user login credentials are used as provisioning server credentials.

**voIpProt.SIP.specialEvent.checkSync.downloadCallList**

0 (default) - The phone does not download the call list for the user after receiving a checksync event in the NOTIFY.
1 - The phone downloads the call list for the user after receiving a checksync event in the NOTIFY.

### Remotely Logging Out Users

Note that if an unexpected reboot occurs while a user is logged in, the user is not logged out and the phone returns to the user profile after reboot.

If a user is not logged out from a phone and other users are not prevented from logging in, the user can ask the administrator to log out remotely. Administrators can log out a user remotely with a checksync event in the NOTIFY by setting the parameter `profileLogout=remote`.

### User Profile Authentication

You can authenticate users with phone-based or server-based authentication methods.

Phone-based authentication authenticates credentials entered by the user against the credentials in the `<user>.cfg` file. Server-based authentication passes user credentials to the provisioning server for authentication.

### User Profile Server Authentication

Instead of phone-based authentication of user profiles, you can authenticate user profiles using a server.

When you enable server authentication, you set up user accounts on the provisioning server and each user can authenticate their phone by entering correct server credentials.

The phone downloads log files (`app.log` and `boot.log`) from the generic profile on the provisioning server regardless of user logins.

### Create a Generic Profile Using Server Authentication

Create a generic profile and generic credentials on the provisioning server when a user isn’t logged into the phone.

If you enable server authentication of user profiles, the following parameters don’t apply and you don’t need to configure them:

- `prov.login.defaultUser`
- prov.login.defaultPassword
- prov.login.defaultOnly
- prov.login.localPassword
- prov.login.localPassword.hashed

Procedure

1. On the server, create an account and directory for the generic profile (for example, Generic_Profile).
2. In the Generic_Profile directory, create a configuration file for a generic profile the phone uses by default (for example, genericprofile.cfg).
3. In genericprofile.cfg, include registration and server details and set all phone feature parameters.

   You must set the following parameters to use server authentication:
   - prov.login.enabled="1"
   - prov.login.useProvAuth="1"
   - prov.login.persistent="1"

   Note: If you enable prov.login.enabled=1 and don't enable prov.login.useProvAuth=0, users are authenticated by a match with credentials you store in the user configuration file <user>.cfg.

4. Create a master configuration file 000000000000.cfg for all the phones, or a <MACAddress>.cfg for each phone, and add genericprofile.cfg to the CONFIG_FILES field.
5. Set the provisioning server address and provisioning server user name and password credentials for the generic user account on the phone at Settings > Advanced > Provisioning Server and inform users of their user profile credentials.

The following override files upload to the generic profile directory:
- Log files
- Local interface settings
- System web interface settings
- Call logs
- Contact directory file

Create a User Profile Using Server Authentication

Create a user profile in the Home directory of each user with a user-specific configuration file that you store on the provisioning server with a unique name as well as user-specific files such as settings, directory, and call lists.

When a user logs in with credentials, the phone downloads the user profile from the provisioning server. When the user logs out, the phone downloads the default user profile using the generic credentials.

Procedure

1. On the server, create an account and a directory for each user (for example, User1 and User2).
2. In each user directory, create a configuration file for each user (for example, User1.cfg and User2.cfg), that contains the user’s registration details and feature settings.
The following override files upload to the generic profile account on the server:

- Log files
- System web interface settings

The following override files upload to the user profile account on the server:

- Local interface settings
- Contact directory file

**User Profile Phone Authentication**

You can create default credentials and authenticate user profiles without using a server.

**Create Default Credentials and a Profile for a Phone**

You can choose to define default credentials for a phone, which the phone uses to automatically log itself in each time an actual user logs out or the phone restarts or reboots.

When the phone logs itself in using the default login credentials, a default phone profile displays, and users retain the option to log in and view their personal settings.

You can create a new phone configuration file for the default profile, then add and set the attributes for the feature. Or, you can update an existing phone configuration file to include the user login parameters you want to change.

**Important:** Poly recommends that you create a single default user password for all users.

**Procedure**

1. Add the `prov.login*` parameters you want to use to your configuration.
2. Set values for the user login parameters and save.

**Create a User Configuration File**

Create a configuration file for each user that you want to enable to log in to the phone.

Some things to note about user configuration files:

- If a user updates their password or other user-specific settings on the phone, the updates are stored in `<user>-phone.cfg`, not `<MACaddress>-phone.cfg`.
- If a user updates their contact directory while logged in to a phone, the updates are stored in `<user>-directory.xml`.
- Directory updates display each time the user logs in to a phone. For certain phones, an up-to-date call lists history is defined in `<user>-calls.xml`. This list is retained each time the user logs in to their phone.

The following list shows configuration parameter precedence (from first to last) for a phone with the user profile feature enabled:

1. `<user>-phone.cfg`
2. System web interface
3. Configuration files listed in the master configuration file (including `<user>.cfg`)
4. Default values
**Note:** To convert a phone-based deployment to a user-based deployment, copy the `<MACaddress>-phone.cfg` file to `<user>-phone.cfg` and copy `phoneConfig<MACaddress>.cfg` to `<user>.cfg`.

**Procedure**

1. On the provisioning server, create a user configuration file for each user. Specify the user’s login ID in the name of the file.
   
   For example, if the user’s login ID is `user100`, name the user configuration file `user100.cfg`.
2. In each `<user>.cfg` file, you must add and set values for the user’s login password.
3. Optional: Add and set values for any user-specific parameters you want to add:
   
   • Registration details, such as the number of lines the profile displays and line labels
   • Feature settings, such as microbrowser settings

**Caution:** If you add optional user-specific parameters to `<user>.cfg`, only add parameters that don’t cause the phone to restart or reboot when the parameter is updated.
Network Configuration

Topics:

- Two-Way Active Measurement Protocol
- 3GPP Technical Specifications
- Technical Report-069
- Advice of Charge
- Enhanced IPv4 ICMP Management
- Real-Time Transport Protocol (RTP) Ports
- Network Address Translation (NAT)
- Server Redundancy
- DNS SIP Server Name Resolution
- Static DNS Cache
- IP Type-of-Service
- SIP Instance Support
- Provisional Polling of Phones
- SIP Subscription Timers
- Incoming Network Signaling Validation
- Session Traversal Utilities for NAT (STUN)
- GZIP Encoding of SIP INFO Messages
- DHCP IP Address
- DHCP IP Address Cache
- Wireless Network Connectivity (Wi-Fi)
- Configuring Bluetooth

Poly UC Software enables you to make custom network configurations.

Two-Way Active Measurement Protocol

UC Software supports Two-Way Active Measurement Protocol (TWAMP), which is RFC 5357 compliant, to check network performance by measuring the round-trip time between two devices using TWAMP protocols.

TWAMP defines the following protocols:

- TWAMP Control protocol, which uses TCP.
- TWAMP Test protocol, which uses UDP.
TWAMP Limitations

TWAMP includes the following limitations:

- TWAMP Control and Test protocols only support unauthenticated mode
- A maximum of 10 clients can establish a connection with the server
- The server is limited to handle a maximum of 10 sessions per client

Two-Way Active Measurement Protocol Configuration Parameters

The following list includes the new or modified parameters for the two-way active measurement protocol feature.

**feature.twamp.enabled**

0 (default) - Disable TWAMP protocol support.
1 - Enable TWAMP protocol support.

**twamp.port.udp.PortRangeEnd**

Set the TWAMP UDP session max port range value.
60000 (default)
1024 - 65486

**twamp.port.udp.PortRangeStart**

Set the TWAMP UDP session start port range value.
40000 (default)
1024 - 65485

**twamp.udp.maxSession**

Set the maximum UDP session supported by TWAMP.
1 (default)
1 - 10

3GPP Technical Specifications


In addition, Poly phones provide partial or complete support for the following RFCs:

- RFC 3327
- RFC 3608
- RFC 3680
- RFC 6665
- RFC 6228
CCX phones support the following IMS feature enhancements:

- The call waiting ring-back tone plays to inform users that a call is waiting at the far end.
- The SIP Response Code 199 (defined in RFC 6228) is supported.
- The Path extension header field in the SIP Register request message allows accumulating and transmitting the list of proxies between a user agent and Registrar server.
- The caller phone can support the p-early-media SIP header that determines whether the caller phone should play a network-provided media or its own media as a ring back tone.
- The VQMon messages generated by the phone can contain service route information in SIP route headers.
- In a NAT network, a phone may need to send keep-alive messages to maintain the IP addresses mapping in the NAT table.

### 3GPP Technical Specifications Parameters

Use the 3GPP parameters in the following list to configure IP Multimedia Subsystem (IMS) features.

**nat.keepalive.tcp.payload**

Configure a customizable string as the payload of a TCP keep-alive message. The string value cannot be blank.

CRLF

**nat.keepalive.udp.payload**

Configure a customizable string as the payload of a UDP keep-alive message. You can leave the string value blank to configure an empty payload.

CRLF

**reg.x.header.pearlymedia.support**

0 (Default) - The p-early-media header is not supported on the specified line registration.
1 - The p-early-media header is supported by the specified line registration.

**reg.X.insertOBPAddressInRoute**

1 (Default) - The outbound proxy address is added as the topmost route header.
0 - The outbound proxy address is not added to the route header.
**reg.x.path**
0 (Default) - The path extension header field in the Register request message is not supported for the specific line registration.
1 - The phone supports and provides the path extension header field in the Register request message for the specific line registration.

**reg.x.regevent**
0 (default) - The phone is not subscribed to registration state change notifications for the specific phone line.
1 - The phone is subscribed to registration state change notifications for the specific phone line.
This parameter overrides the global parameter voIpProt.SIP.regevent.

**reg.x.rejectNDUBInvite**
Specify whether or not the phone accepts a call for a particular registration in case of a Network Determined User Busy (NDUB) event advertised by the SIP server.
0 (Default) - If an NDUB event occurs, the phone does not reject the call.
1 - If an NDUB event occurs, the phone rejects the call with a 603 Decline response code.

**reg.x.server.y.specialInterop**
Specify the server-specific feature set for the line registration.
Standard (default), GENBAND, ALU-CTS, ocs2007r2, lcs2005

**voice.qualityMonitoring.processServiceRoute.enable**
0 (Default) - The VQMon messages generated by the phone do not contain service route information in SIP route headers.
1 - The VQMon messages generated by the phone contain service route information, if available, in SIP route headers.
Change causes system to restart or reboot.

**voIpProt.SIP.header.pEarlyMedia.support**
0 (Default) - The p-early-media header is not supported by the caller phone.
1 - The p-early-media header is supported by the caller phone.

**voIpProt.SIP.IMS.enable**
This parameter applies to all registered or unregistered SIP lines on the phone.
0 (Default) - The phone does not support IMS features introduced in UC Software 5.5.0.
1 - The phone supports IMS features introduced in UC Software 5.5.0.

**voIpProt.SIP.regevent**
0 (default) - The phone is not subscribed to registration state change notifications for all phone lines.
1 - The phone is subscribed to registration state change notifications for all phone lines.
   This parameter is overridden by the per-phone parameter reg.x.regevent.

**voIpProt.SIP.rejectNDUBInvite**

Specify whether or not the phone accepts a call for all registrations in case of a Network
Determined User Busy (NDUB) event advertised by the SIP server.
0 (Default) - If an NDUB event occurs, the phone does not reject the call for all line registrations.
1 - If an NDUB event occurs, the phone rejects the call with a 603 Decline response code for all
line registrations.

**voIpProt.SIP.supportFor199**

Determine support for the 199 response code. For details on the 199 response code, see RFC 6228.
0 (Default) - The phone does not support the 199 response code.
1- The phone supports the 199 response code.

**Technical Report-069**

Technical Report-069 (TR-069) enables you to remotely manage end-user devices.

As a bidirectional SOAP/HTTP-based protocol, TR-069 enables secure communication between Auto
Configuration Servers (ACS) and Poly phones. Using TR-069, you can remotely configure and manage
Poly phones by provisioning systems that comply with TR-069 technical specification.

**TR-069 Parameters**

Poly provides parameters for the TR-104 and TR-106 data models that support provisioning of TR-069-enabled devices by an Auto-Configuration Server (ACS).

TR-104 is a parameter data model for VoIP-only devices, and TR-106 is a parameter data model for all
TR-069-enabled devices.

**device.feature.tr069.enabled**

0 (default) - Disables TR-069 feature
1 - Enables TR-069 feature

**device.feature.tr069.enabled.set**

0 (default) - Disabled
1 - Enabled

**device.tr069.acs.password**

Sets the TR-069 ACS server password used to authenticate the phone.
Null (default)
device.tr069.acs.url
Sets the URL for the TR-069 ACS server.
Null (default)
URL (256 maximum characters)

device.tr069.acs.username
Sets the TR-069 ACS server user name used to authenticate the phone.
PlcmSpip (default)
String (256 maximum characters)

device.tr069.cpe.password
Specifies the TR-069 CPE password, which authenticates a connection request from the ACS server.
Null (default)
String (256 maximum characters)

device.tr069.cpe.username
Specifies the TR-069 CPE user name, which authenticates a connection request from the ACS server.
PlcmSpip (default)
String (256 maximum characters)

device.tr069.periodicInform.enabled
Indicates whether the CPE must periodically send CPE information to ACS using the Inform method call.
0 (default) - Periodic Inform call is disabled.
1 - Periodic Inform call is enabled.

device.tr069.periodicInform.interval
Specifies the time interval in seconds in which the CPE must attempt to connect with the ACS to send CPE information if device.tr069.periodicInform.enabled ="1".
18000 (default)
0 to 36000

device.tr069.upgradesManaged.enabled
Indicates whether the ACS manages image upgrades for the phone or not.
0 (default) - The phone uses ACS or provisioning server for upgrade.
1 - The phone upgrades only from the ACS server.
log.level.change.tr069  
Sets the log levels for the TR-069 feature.

4 (default)
0 - 6

Configuring TR-069  
You can configure the TR-069 feature through the phone menu, Web Configuration Utility, or configuration parameters on a central server.

You can configure Poly phones with an ACS server, including user name and password, using DHCP Option 43 for IPv4 and DHCP Option 17 for IPv6.

Configure TR-069 Settings on the Phone Menu  
You can configure TR-069 settings on the phone menu.

Procedure
1. Go to Settings > Advanced > Administration Settings > Network Configuration.
2. Select TR-069, and select Enabled.
3. In the TR069 Menu, select ACS Configuration and enter values for the following settings:
   • URL
   • Username
   • Password
   • Periodic Inform
   • Inform Interval
4. In Phone/CPE Configuration, configure a user name and password.
5. In Upgrade Management, select Enable or Disable.

Configure TR-069 from the Web Configuration Utility  
You can configure TR-069 from the Web Configuration Utility.

Procedure
» In the Web Configuration Utility, navigate to Settings > Provisioning Server > TR-069 Menu.

Map TR-106 Parameters to Poly Parameters  
The data model TR-106 defines the TR-069 ACS parameter details.

The parameters listed as 'Internal Value' are not directly mapped to a configuration parameter on the phone, and the phone generates these values dynamically to provide to the ACS server.

The following table lists the TR-106 parameters and their corresponding Poly parameters.
## TR-106 Parameters to Poly Parameters

<table>
<thead>
<tr>
<th>TR-106 ACS parameter names</th>
<th>Parameter (Poly parameter names)</th>
<th>Writable</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Device</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Device.DeviceInfo</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manufacturer</td>
<td>Internal Value</td>
<td>No</td>
</tr>
<tr>
<td>ManufacturerOUI</td>
<td>Internal Value</td>
<td>No</td>
</tr>
<tr>
<td>ModelName</td>
<td>Internal Value</td>
<td>No</td>
</tr>
<tr>
<td>ProductClass</td>
<td>Internal Value</td>
<td>No</td>
</tr>
<tr>
<td>SerialNumber</td>
<td>Internal Value</td>
<td>No</td>
</tr>
<tr>
<td>HardwareVersion</td>
<td>Internal Value</td>
<td>No</td>
</tr>
<tr>
<td>SoftwareVersion</td>
<td>Internal Value</td>
<td>No</td>
</tr>
<tr>
<td>UpTime</td>
<td>Internal Value</td>
<td>No</td>
</tr>
<tr>
<td><strong>Device.ManagementServer</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>URL</td>
<td>device.tr069.acs.url</td>
<td>Yes</td>
</tr>
<tr>
<td>Username</td>
<td>device.tr069.acs.username</td>
<td>Yes</td>
</tr>
<tr>
<td>Password</td>
<td>device.tr069.acs.password</td>
<td>Yes</td>
</tr>
<tr>
<td>PeriodicInformEnable</td>
<td>device.tr069.periodicInform.enabled</td>
<td>Yes</td>
</tr>
<tr>
<td>PeriodicInformInterval</td>
<td>device.tr069.periodicInform.interval</td>
<td>Yes</td>
</tr>
<tr>
<td>ConnectionRequestURL</td>
<td>Internal Value</td>
<td>No</td>
</tr>
<tr>
<td>ConnectionRequestUsername</td>
<td>device.tr069.cpe.username</td>
<td>Yes</td>
</tr>
<tr>
<td>ConnectionRequestPassword</td>
<td>device.tr069.cpe.password</td>
<td>Yes</td>
</tr>
<tr>
<td>UpgradesManaged</td>
<td>device.tr069.upgradesManaged.enabled</td>
<td>Yes</td>
</tr>
<tr>
<td>STUNServerAddress</td>
<td>tcpIpApp.ice.stun.server</td>
<td>Yes</td>
</tr>
<tr>
<td>STUNServerPort</td>
<td>tcpIpApp.ice.stun.udpPort</td>
<td>Yes</td>
</tr>
<tr>
<td>STUNUsername</td>
<td>tcpIpApp.ice.username</td>
<td>Yes</td>
</tr>
<tr>
<td>STUNPassword</td>
<td>tcpIpApp.ice.password</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Device.LAN</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IPAddress</td>
<td>Internal Value</td>
<td>No</td>
</tr>
</tbody>
</table>
TR-106 ACS parameter names | Parameter (Poly parameter names) | Writable
---|---|---
SubnetMask | Internal Value | No
DNSServers | Internal Value | No
MACAddress | Internal Value | No
MACAddressOverride | Internal Value | No

**Map TR-104 Parameters to Poly Parameters**

The data model TR-104 defines the TR-069 ACS parameter details.
The parameters listed as ‘Internal Value’ are not directly mapped to a configuration parameter on the phone and the phone generates these values dynamically to provide to the ACS server.
The following table lists the TR-104 parameters and their corresponding Poly parameters.

<table>
<thead>
<tr>
<th>TR-104 ACS parameter names</th>
<th>CPE Parameter (Poly parameter names)</th>
<th>Writable</th>
</tr>
</thead>
<tbody>
<tr>
<td>VoiceService.{i}.VoiceProfile.{i}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DigitMap</td>
<td>dialplan.digitmap</td>
<td>Yes</td>
</tr>
<tr>
<td>VoiceService.{i}.VoiceProfile.{i}.SIP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RegistrarServer</td>
<td>voIPProt.server.X.address</td>
<td>Yes</td>
</tr>
<tr>
<td>RegistrarServerPort</td>
<td>voIPProt.server.X.port</td>
<td>Yes</td>
</tr>
<tr>
<td>OutboundProxy</td>
<td>voIPProt.SIP.outboundProxy.address</td>
<td>Yes</td>
</tr>
<tr>
<td>OutboundProxyPort</td>
<td>voIPProt.SIP.outboundProxy.port</td>
<td>Yes</td>
</tr>
<tr>
<td>RegisterExpires</td>
<td>voIPProt.server.X.expires</td>
<td>Yes</td>
</tr>
<tr>
<td>RegistersMinExpires</td>
<td>voIPProt.server.X.expires.overlap</td>
<td>Yes</td>
</tr>
<tr>
<td>RegisterRetryInterval</td>
<td>voIPProt.server.X.retryTimeOut</td>
<td>Yes</td>
</tr>
<tr>
<td>VoiceService.{i}.VoiceProfile.{i}.SIP.EventSubscribe.{i}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ExpireTime</td>
<td>voIPProt.server.X.subscribe.expires</td>
<td>Yes</td>
</tr>
<tr>
<td>VoiceService.{i}.VoiceProfile.{i}.H323</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gatekeeper</td>
<td>voIPProt.server.H323.X.address</td>
<td>Yes</td>
</tr>
<tr>
<td>GatekeeperPort</td>
<td>voIPProt.server.H323.X.port</td>
<td>Yes</td>
</tr>
<tr>
<td>VoiceService.{i}.VoiceProfile.{i}.RTP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TR-104 ACS parameter names</td>
<td>CPE Parameter (Poly parameter names)</td>
<td>Writable</td>
</tr>
<tr>
<td>----------------------------</td>
<td>--------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>LocalPortMin</td>
<td>tcpIpApp.port.rtp.mediaPortRangeStart</td>
<td>Yes</td>
</tr>
<tr>
<td>LocalPortMax</td>
<td>tcpIpApp.port.rtp.mediaPortRangeEnd</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>VoiceService.{i}.VoiceProfile.{i}.RTP.SRTP</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enable</td>
<td>sec.srtp.enable</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>VoiceService.{i}.VoiceProfile.{i}.ButtonMap.Button.{i}</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ButtonName</td>
<td>softkey.X.label</td>
<td>Yes</td>
</tr>
<tr>
<td>FacilityAction</td>
<td>softkey.X.action</td>
<td>Yes</td>
</tr>
<tr>
<td>UserAccess</td>
<td>softkey.X.enable</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>VoiceService.{i}.VoiceProfile.{i}.Line.{i}</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DirectoryNumber</td>
<td>reg.X.address</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>VoiceService.{i}.VoiceProfile.{i}.Line.{i}.SIP</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AuthUserName</td>
<td>reg.X.auth.userId</td>
<td>Yes</td>
</tr>
<tr>
<td>AuthPassword</td>
<td>reg.X.auth.password</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>VoiceService.{i}.VoiceProfile.{i}.Line.{i}.CallingFeatures</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CallForwardUnconditionalEnable</td>
<td>reg.X.fwdStatus</td>
<td>Yes</td>
</tr>
<tr>
<td>CallForwardUnconditionalNumber</td>
<td>reg.X.fwdContact</td>
<td>Yes</td>
</tr>
<tr>
<td>CallForwardOnBusyEnable</td>
<td>reg.X.fwd.busy.status</td>
<td>Yes</td>
</tr>
<tr>
<td>CallForwardOnBusyNumber</td>
<td>reg.X.fwd.busy.contact</td>
<td>Yes</td>
</tr>
<tr>
<td>CallForwardOnNoAnswerEnable</td>
<td>reg.X.fwd.noanswer.status</td>
<td>Yes</td>
</tr>
<tr>
<td>CallForwardOnNoAnswerNumber</td>
<td>reg.X.fwd.noanswer.contact</td>
<td>Yes</td>
</tr>
<tr>
<td>CallForwardOnNoAnswerRingCount</td>
<td>reg.X.fwd.noanswer.ringCount</td>
<td>Yes</td>
</tr>
<tr>
<td>DoNotDisturbEnable</td>
<td>divert.dnd.X.enabled</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Supported TR-069 Remote Procedure Call (RPC) Methods

The following table lists the supported RPC methods.

### RPC Methods

<table>
<thead>
<tr>
<th>RPC Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetRPCMethods</td>
<td>Discovers the set of methods supported by the phone.</td>
</tr>
<tr>
<td>SetParameterValues</td>
<td>Modifies the value of one or more phone parameters.</td>
</tr>
<tr>
<td>GetParameterValues</td>
<td>Obtains the value of one or more phone parameters.</td>
</tr>
<tr>
<td>GetParameterNames</td>
<td>Discovers the parameters accessible on a particular phone.</td>
</tr>
<tr>
<td>GetParameterAttributes</td>
<td>Reads the attributes associated with one or more phone parameters.</td>
</tr>
<tr>
<td>SetParameterAttributes</td>
<td>Modifies attributes associated with one or more phone parameters.</td>
</tr>
<tr>
<td>Reboot</td>
<td>Reboots the phone.</td>
</tr>
<tr>
<td>Download</td>
<td>Causes the phone to download a specified file from the designated location.</td>
</tr>
<tr>
<td></td>
<td>Supported file types for download:</td>
</tr>
<tr>
<td></td>
<td>Firmware Image</td>
</tr>
<tr>
<td></td>
<td>Configuration File</td>
</tr>
<tr>
<td>FactoryReset</td>
<td>Resets the phone to its factory default state.</td>
</tr>
<tr>
<td>TransferComplete</td>
<td>Informs the ACS of the completion (either successful or unsuccessful) of a</td>
</tr>
<tr>
<td></td>
<td>file transfer initiated by an earlier Download or Upload method call.</td>
</tr>
<tr>
<td>AddObject</td>
<td>Adds a new instance of an object defined on the phone.</td>
</tr>
<tr>
<td>DeleteObject</td>
<td>Removes a particular instance of an object.</td>
</tr>
</tbody>
</table>

**Advice of Charge**

In an IP Multimedia Subsystem (IMS) environment, Poly phones support the Advice of Charge (AoC) feature as defined in Technical Specification (TS) **24.647 version 9.1.0 Release 9**.

You can enable Poly phones to display call charges information, which can include:

- Call setup charge and call tariff information - Displayed at the beginning of a call.
- Cumulative call cost - Displayed on an ongoing call.
- Complete call cost - Displayed after a call ends.
**Advice of Charge Parameters**

The following parameters configure the Advice of Charge (AoS) feature.
Before configuring AoS parameters, you must set `voIpProt.SIP.IMS.enable` to 1.

**voIpProt.SIP.aoc.enable**
- 0 (Default) - The phone does not display call charge information.
- 1 - The phone displays call charge information.

**feature.adviceOfCharge.allowAudioNotification**
- 0 (Default) - There is no audio beep sound when the call charges information is updated on the phone display.
- 1 - The phone gives an audio beep when the call charges information is updated on the phone display.

**Enhanced IPv4 ICMP Management**

Poly phones support IPv4 by enabling the phone to ignore Internet Control Message Protocol (ICMP) redirect requests for an alternate path from the router or gateway.

**IPv4 Parameters**

You can configure IPv4 using parameters listed below.

**device.icmp.ipv4IcmpIgnoreRedirect.set**
- 0 (default) - The phone does not allow to use `device.icmp.ipv4IcmpIgnoreRedirect` parameter to configure Enhanced IPv4 ICMP Management feature.
- 1 - The phone allows to use `device.icmp.ipv4IcmpIgnoreRedirect` parameter to configure Enhanced IPv4 ICMP Management feature.

**device.icmp.ipv4IcmpIgnoreRedirect**
- 1 (default) - The phone ignores ICMP redirect requests for an alternate path from the router or gateway.
- 0 - The phone allows ICMP redirects.

**Real-Time Transport Protocol (RTP) Ports**

You can configure RTP ports for your environment in the following ways:

- Filter incoming packets by IP address or port.
- Reject packets arriving from a non-negotiated IP address, an unauthorized source, or non-negotiated port for greater security.
• Enforce symmetric port operation for RTP packets. When the source port is not set to the negotiated remote sink port, arriving packets are rejected.
• Fix the phone's destination transport port to a specified value regardless of the negotiated port.
  This is useful for communicating through firewalls. When you use a fixed transport port, all RTP traffic is sent to and arrives on that specified port. Incoming packets are sorted by the source IP address and port, which allows multiple RTP streams to be multiplexed.
• Specify the phone's RTP port range.
  Since the phone supports conferencing and multiple RTP streams, the phone can use several ports concurrently. Consistent with RFC 1889, 3550, and 3551, the next-highest odd-numbered port is used to send and receive RTP.

RTP Ports Parameters
Use the parameters in the following list to configure RTP packets and ports.

```
tcpIpApp.port.rtp.feccPortRange.enable
  0 (default) – Use the Open SIP far-end camera control media port range.
  1 - Use the far-end camera control port range configuration for Open SIP-registered lines.

tcpIpApp.port.rtp.feccPortRangeEnd
  Specify the far-end camera control port range end port for Open SIP registrations.
  2419 (default)
  1024 - 65486

tcpIpApp.port.rtp.feccPortRangeStart
  Specify the far-end camera control port range start port for Open SIP registrations.
  2372 (default)
  1024 – 65486

tcpIpApp.port.rtp.filterByIp1
  IP addresses can be negotiated through the SDP or H.323 protocols.
  1 (Default) - Phone rejects RTP packets that arrive from non-negotiated IP addresses.
  Change causes system to restart or reboot.

tcpIpApp.port.rtp.filterByPort1
  Ports can be negotiated through the SDP protocol.
  0 (Default)
  1 - Phone rejects RTP packets arriving from (sent from) a non-negotiated port.
  Change causes system to restart or reboot.
```
tcpIpApp.port.rtp.forceSend1

Send all RTP packets to, and expect all RTP packets to arrive on, this port. Range is 0 to 65535.
0 (Default) - RTP traffic is not forced to one port.
Both tcpIpApp.port.rtp.filterByIp and tcpIpApp.port.rtp.filterByPort must be set to 1.
Change causes system to restart or reboot.

tcpIpApp.port.rtp.mediaPortRangeEnd

Determines the maximum supported end range of audio ports. Range is 1024 to 65485.
2269 (Default)
Change causes system to restart or reboot.

tcpIpApp.port.rtp.mediaPortRangeStart1

Set the starting port for RTP port range packets. Use an even integer ranging from 1024 to 65440.
2222 (Default)
Each call increments the port number +2 to a maximum of 24 calls after the value resets to the starting point. Because port 5060 is used for SIP signaling, ensure that port 5060 is not within this range when you set this parameter. A call that attempts to use port 5060 has no audio.
Change causes system to restart or reboot.

tcpIpApp.port.rtp.videoPortRange.enable

Specifies the range of video ports.
0 - Video ports are chosen within the range specified by tcpIpApp.port.rtp.mediaPortRangeStart and tcpIpApp.port.rtp.mediaPortRangeEnd.
1 - Video ports are chosen from the range specified by tcpIpApp.port.rtp.videoPortRangeStart and tcpIpApp.port.rtp.videoPortRangeEnd.
Generic = 0 (Default)

tcpIpApp.port.rtp.videoPortRangeEnd

Determines the maximum supported end range of video ports. Range is 1024 to 65535.
2319 (Default)
Change causes system to restart or reboot.

tcpIpApp.port.rtp.videoPortRangeStart

Determines the start range for video ports. Range is 1024 to 65486.
2272 (Default)
Used only if value of tcpIpApp.port.rtp.videoPortRange.enable is 1.
Change causes system to restart or reboot.
Network Address Translation (NAT)

Network Address Translation (NAT) enables a local area network (LAN) to use one set of IP addresses for internal traffic and another set for external traffic.

The phone’s signaling and RTP traffic use symmetric ports. Note that the source port in transmitted packets is the same as the associated listening port used to receive packets.

Network Address Translation Parameters

You can configure the external IP addresses and ports used by the NAT on the phone’s behalf on a per-phone basis.

Use the parameters in the following list to configure NAT.

nat.ip

Specifies the IP address to advertise within SIP signaling. This should match the external IP address used by the NAT device.

- Null (default)
- IP address

Change causes system to restart or reboot.

nat.keepalive.interval

The keep-alive interval in seconds. Sets the interval at which phones send a keep-alive packet to the gateway/NAT device to keep the communication port open so that NAT can continue to function. If Null or 0, the phone does not send out keep-alive messages.

- 0 (default)
- 0 - 3600

nat.mediaPortStart

The initially allocated RTP port. Overrides the value set for tcpIpApp.port.rtp.mediaPortRangeStart parameter.

- 0 (default)
- 0 - 65440

Change causes system to restart or reboot.

nat.signalPort

The port used for SIP signaling. Overrides the voIpProt.local.port parameter.

- 0 (default)
- 1024 - 65535
Server Redundancy

Server redundancy is often required in VoIP deployments to ensure continuity of phone service if, for example, the call server is taken offline for maintenance, the server fails, or the connection between the phone and the server fails.

Poly phones support failover and fallback server redundancy types. In some cases, you can deploy a combination of the two server redundancy types. Consult your SIP server provider for recommended methods of configuring phones and servers for failover configuration.

**Note:** The concurrent failover/fallback feature is not compatible with Microsoft environments.

For more information, see Technical Bulletin 5844: SIP Server Fallback Enhancements on Polycom Phones and Technical Bulletin 66546: Configuring Optional Re-Registration on Failover Behavior.

Server Redundancy Parameters

Use the parameters in the following list to set up server redundancy for your environment.

**reg.x.auth.optimizedInFailover**

Set the destination for the first new SIP request when failover occurs.

- 0 (default) - The SIP request is sent to the server with the highest priority in the server list.
- 1 - The SIP request is sent to the server that sent the proxy authentication request.

**reg.x.outboundProxy.failOver.failBack.mode**

The mode for failover failback (overrides reg.x.server.y.failOver.failBack.mode).

- duration (default) - The phone tries the primary server again after the time specified by reg.x.outboundProxy.failOver.failBack.timeout expires.
- newRequests - All new requests are forwarded first to the primary server regardless of the last used server.
- DNSTTL - The phone tries the primary server again after a timeout equal to the DNS TTL you configured for the server the phone is registered to.

**reg.x.outboundProxy.failOver.failBack.timeout**

3600 (default) - The time to wait (in seconds) before failback occurs (overrides reg.x.server.y.failOver.failBack.timeout).

- 0, 60 to 65535 - The phone does not fail back until a failover event occurs with the current server.

**reg.x.outboundProxy.failOver.failRegistrationOn**

1 (default) - The global and per-line reRegisterOn parameter is enabled and the phone silently invalidates an existing registration.

- 0 - The global and per-line reRegisterOn parameter is enabled and existing registrations remain active.
**reg.x.outboundProxy.failOver.onlySignalWithRegistered**

1 (default) - The global and per-line reRegisterOn and failRegistrationOn parameters are enabled, no signaling is accepted from or sent to a server that has failed until failback is attempted or failover occurs.

0 - The global and per-line reRegisterOn and failRegistrationOn parameters are enabled, signaling is accepted from and sent to a server that has failed.

**reg.x.outboundProxy.failOver.reRegisterOn**

This parameter overrides reg.x.server.y.failOver.reRegisterOn.

0 (default) - The phone won't attempt to register with the secondary server.

1 - The phone attempts to register with (or via, for the outbound proxy scenario), the secondary server.

**reg.x.outboundProxy.port**

The port of the SIP server to which the phone sends all requests.

0 - (default)

1 to 65535

**reg.x.outboundProxy.transport**

The transport method the phone uses to communicate with the SIP server.

DNSnaptr (default)

DNSnaptr, TCPpreferred, UDPOnly, TLS, TCPOnly

**voIpProt.server.x.failOver.failBack.mode**

Specify the failover failback mode.

duration (default) - The phone tries the primary server again after the time specified by voIpProt.server.x.failOver.failBack.timeout

newRequests - All new requests are forwarded first to the primary server regardless of the last used server.

DNSTTL - The phone tries the primary server again after a timeout equal to the DNS TTL configured for the server that the phone is registered to.

registration - The phone tries the primary server again when the registration renewal signaling begins.

**voIpProt.server.x.failOver.failBack.timeout**

If voIpProt.server.x.failOver.failBack.mode is set to duration, this is the time in seconds after failing over to the current working server before the primary server is again selected as the first server to forward new requests. Values between 1 and 59 result in a timeout of 60. 0 means do not fail-back until a fail-over event occurs with the current server.

3600 (default)

0, 60 to 65535
voIpProt.server.x.failOver.failRegistrationOn

1 (default) - When set to 1, and the global or per-line reRegisterOn parameter is enabled, the phone silently invalidates an existing registration (if it exists), at the point of failing over.

0 - When set to 0, and the global or per-line reRegisterOn parameter is enabled, existing registrations remain active. This means that the phone attempts failback without first attempting to register with the primary server to determine if it has recovered.

voIpProt.server.x.failOver.onlySignalWithRegistered

1 (default) - When set to 1, and the global or per-line reRegisterOn and failRegistrationOn parameters are enabled, no signaling is accepted from or sent to a server that has failed until failback is attempted or failover occurs. If the phone attempts to send signaling associated with an existing call via an unregistered server (for example, to resume or hold a call), the call ends. No SIP messages are sent to the unregistered server.

0 - When set to 0, and the global or per-line reRegisterOn and failRegistrationOn parameters are enabled, signaling is accepted from and sent to a server that has failed (even though failback hasn’t been attempted or failover hasn’t occurred).

voIpProt.server.x.failOver.reRegisterOn

0 (default) - When set to 0, the phone won’t attempt to register with the second.

1 - When set to 1, the phone attempts to register with (or by, for the outbound proxy scenario), the secondary server. If the registration succeeds (a 200 OK response with valid expires), signaling proceeds with the secondary server.

DNS SIP Server Name Resolution

If a DNS name is given for a proxy/registrar address, the IP addresses associated with that name is discovered as specified in RFC3263.

If a port is given, the only lookup is an A record. If no port is given, NAPTR and SRV records are tried before falling back on A records if NAPTR and SRV records return no results. If no port is given, and none is found through DNS, port 5060 is used. If the registration type is TLS, port 5061 is used.

Caution: Failure to resolve a DNS name is treated as signaling failure that causes a failover.

The following configuration causes the phone to build an SRV request based on the address you provide, including all subdomains. Use the format:

- voIpProt.SIP.outboundProxy.address="sip.example.com"
- voIpProt.SIP.outboundProxy.port="0"

This SRV request produces a list of servers ordered by weight and priority, enabling you to specify subdomains for separate servers, or you can create partitions of the same system. Please note that while making SRV queries and transport is configured as TCP, the phone adds the prefix <_service._proto.> to the configured address/FQDN but does not remove the sub-domain prefix, for example sip.example.com becomes _sip._tcp.sip.example.com. A single SRV query can be resolved into many different servers, session border controllers (SBCs), or partitions ordered by weight and priority, for example, voice.sip.example.com and video.sip.example.com. Alternatively, use DNS NAPTR to discover what services are available at the root domain.
Customer Phone Configuration

The phones at the customer site are configured as follows:

• Server 1 (the primary server) is configured with the address of the service provider call server. The IP address of the server(s) is provided by the DNS server, for example: reg.1.server.1.address=voipserver.serviceprovider.com .

• Server 2 (the fallback server) is configured to the address of the router/gateway that provides the fallback telephony support and is on-site, for example: reg.1.server.2.address=172.23.0.1 .

Caution: Be careful when using multiple servers per registration. It is possible to configure the phone for more than two servers per registration but ensure that the phone and network load generated by registration refresh of multiple registrations does not become excessive. This is of particular concern when a phone has multiple registrations with multiple servers per registration and some of these servers are unavailable.

For Outgoing Calls (INVITE Fallback)

At the start of a call, server availability is determined by SIP signaling failure. SIP signaling failure depends on the SIP protocol being used:

• If TCP is used, then the signaling fails if the connection fails or the Send fails.

• If UDP is used, then the signaling fails if ICMP is detected or if the signal times out. If the signaling has been attempted through all servers in the list and this is the last server, then the signaling fails after the complete UDP timeout defined in RFC 3261. If it is not the last server in the list, the maximum number of retries using the configurable retry timeout is used.

Caution: If DNS is used to resolve the address for Servers, the DNS server is unavailable, and the TTL for the DNS records has expired, the phone attempts to contact the DNS server to resolve the address of all servers in its list before initiating a call. These attempts timeout, but the timeout mechanism can cause long delays (for example, two minutes) before the phone call proceeds using the working server. To prevent this issue, long TTLs should be used. Poly recommends deploying an on-site DNS server as part of the redundancy solution.

When the user initiates a call, the phone completes the following steps to connect the call:

1. The phone tries to call the working server.

2. If the working server does not respond correctly to the INVITE, the phone tries and makes a call using the next server in the list (even if there is no current registration with these servers). This could be the case if the Internet connection has gone down, but the registration to the working server has not yet expired.

3. If the second server is also unavailable, the phone tries all possible servers (even those not currently registered) until it either succeeds in making a call or exhausts the list at which point the call fails.
VoIP Server Parameters
The list below describes VoIP server configuration parameters.

voIpProt.server.dhcp.available
0 (default) - Do not check with the DHCP server for the SIP server IP address.
1 - Check with the server for the IP address.
Change causes system to restart or reboot.

voIpProt.server.dhcp.option
The option to request from the DHCP server if voIpProt.server.dhcp.available = 1.
128 (default) to 254
If reg.x.server.y.address is non-Null, it takes precedence even if the DHCP server is available.
Change causes system to restart or reboot.

voIpProt.server.dhcp.type
Type to request from the DHCP server if voIpProt.server.dhcp.available is set to 1.
0 (default) - Request IP address
1 - Request string
Change causes system to restart or reboot.

voIpProt.OBP.dhcpv4.type
Define the type of Outbound Proxy address.
0 (default) - IP address
1 - String
Change causes system to restart or reboot.

voIpProt.OBP.dhcpv4.option
The phone requests for DHCP option 120 and applies the outbound proxy obtained in DHCP to 120 (default)
Change causes system to restart or reboot.

voIpProt.OBP.dhcpv6.option
Define the type of Outbound Proxy address from DHCPv6.
21 (default) - list of domain name
22 - list of IP address
Change causes system to restart or reboot.
Phone Operation for Registration

After the phone has booted up, it registers to all configured servers.

Server 1 is the primary server and supports greater SIP functionality than other servers. For example, SUBSCRIBE/NOTIFY services used for features such as shared lines, presence, and BLF is established only with Server 1.

Upon the registration timer expiry of each server registration, the phone attempts to re-register. If this is unsuccessful, normal SIP re-registration behavior (typically at intervals of 30 to 60 seconds) proceeds and continues until the registration is successful (for example, when the Internet link is again operational). While the primary server registration is unavailable, the next highest priority server in the list serves as the working server. As soon as the primary server registration succeeds, it returns to being the working server.

Note: If `reg.x.server.y.register` is set to 0, the phone does not register to that server. However, the INVITE fails over to that server if all higher priority servers are down.

Recommended Practices for Fallback Deployments

In situations where server redundancy for fallback purpose is used, the following measures should be taken to optimize the solution:

- Deploy an on-site DNS server to avoid long call initiation delays that can result if the DNS server records expire.
- Do not use `OutBoundProxy` configurations on the phone if the `OutBoundProxy` could be unreachable when the fallback occurs.
- Avoid using too many servers as part of the redundancy configuration as each registration generates more traffic.
- Educate users as to the features that are not available when in fallback operating mode.

Note: The concurrent/registration failover/fallback feature is not compatible with Microsoft environments.

Static DNS Cache

Failover redundancy can be used only when the configured IP server hostname resolves (through SRV or A record) to multiple IP addresses.

Unfortunately, the DNS cache cannot always be configured to take advantage of failover redundancy.

You can statically configure a set of DNS NAPTR SRV and/or A records into the phone. You can enter a maximum of 12 record entries for DNS-A, DNS-NAPTR, and DNS-SRV. records.

Support for negative DNS caching as described in RFC 2308 is also provided to allow faster failover when prior DNS queries have returned no results from the DNS server. For more information, see RFC2308.
Configuring Static DNS

If a phone is not configured with a DNS server, when the phone attempts to resolve a hostname within the static DNS cache, it always returns the results from the static cache.

Phones configured with a DNS server behave as follows:

1. The phone makes an initial attempt to resolve a hostname that is within the static DNS cache. For example, a query is made to the DNS if the phone registers with its SIP registrar.
2. If the initial DNS query returns no results for the hostname or cannot be contacted, then the values in the static cache are used for their configured time interval.
3. After the configured time interval has elapsed, a resolution attempt of the hostname again results in a query to the DNS.
4. If a DNS query for a hostname that is in the static cache returns a result, the values from the DNS are used and the statically cached values are ignored.

Static DNS Parameters

Use the following parameters to configure static DNS settings.

**reg.x.address**

The user part (for example, 1002) or the user and the host part (for example, 1002@polycom.com) of the registration SIP URI.

Null (default)

string address

**reg.x.server.y**

Specify the call server used for this registration.

**reg.x.server.y.specialInterop**

Specify the server-specific feature set for the line registration.

All other phones:

Standard (default), GENBAND, ALU-CTS, ocs2007r2, lcs2005

**reg.x.server.y.address**

If this parameter is set, it takes precedence even if the DHCP server is available.

Null (default) - SIP server does not accepts registrations.

IP address or hostname - SIP server that accepts registrations. If not Null, all of the parameters in this list override the parameters specified in voIpProt.server.*.

**reg.x.server.y.expires**

The phone's requested registration period in seconds. The period negotiated with the server may be different. The phone attempts to re-register at the beginning of the overlap period.

3600 - (default)
positive integer, minimum 10

reg.x.server.y.expires.lineSeize
Requested line-seize subscription period.
30 - (default)
0 to 65535

reg.x.server.y.expires.overlap
The number of seconds before the expiration time returned by server x at which the phone should try to re-register. The phone tries to re-register at half the expiration time returned by the server if the server value is less than the configured overlap value.
60 (default)
5 to 65535

reg.x.server.y.failOver.failBack.mode
duration (default) - The phone tries the primary server again after the time specified by reg.x.server.y.failOver.failBack.timeout.
newRequests - All new requests are forwarded first to the primary server regardless of the last used server.
DNSTTL - The phone tries the primary server again after a timeout equal to the DNS TTL configured for the server that the phone is registered to.
registration - The phone tries the primary server again when the registration renewal signaling begins.

Note: This parameter overrides voIpProt.server.x.failOver.failBack.mode.

reg.x.server.y.failOver.failBack.timeout
3600 (default) - The time to wait (in seconds) before failback occurs.
0 - The phone does not fail back until a failover event occurs with the current server.
60 to 65535 - If set to Duration, the phone waits this long after connecting to the current working server before selecting the primary server again.

reg.x.server.y.failOver.failRegistrationOn
1 (default) - The reRegisterOn parameter is enabled, the phone silently invalidates an existing registration (if it exists), at the point of failing over.
0 - The reRegisterOn parameter is disabled, existing registrations remain active.

reg.x.server.y.failOver.onlySignalWithRegistered
1 (default) - Set to this value and reRegisterOn and failRegistrationOn parameters are enabled, no signaling is accepted from or sent to a server that has failed until failback is attempted or failover occurs. If the phone attempts to send signaling associated with an existing
call via an unregistered server (for example, to resume or hold a call), the call ends. No SIP
messages are sent to the unregistered server.

0 - Set to this value and reRegisterOn and failRegistrationOn parameters are enabled,
signaling is accepted from and sent to a server that has failed (even though failback hasn't been
attempted or failover hasn't occurred).

`reg.x.server.y.failOver.reRegisterOn`

0 (default) - The phone does not attempt to register with the secondary server, since the phone
assumes that the primary and secondary servers share registration information.

1 - The phone attempts to register with (or via, for the outbound proxy scenario), the secondary
server. If the registration succeeds (a 200 OK response with valid expires), signaling proceeds
with the secondary server.

Note: This parameter overrides voIpProt.server.x.failOver.reRegisterOn.

`reg.x.server.y.port`

Null (default) - The port of the SIP server does not specify registrations.

0 - The port used depends on `reg.x.server.y.transport`.

1 to 65535 - The port of the SIP server that specifies registrations.

`reg.x.server.y.register`

1 (default) - Calls can not be routed to an outbound proxy without registration.

0 - Calls can be routed to an outbound proxy without registration.

See voIpProt.server.x.register for more information, see SIP Server Fallback Enhancements on
Polycom Phones - Technical Bulletin 5844 on Polycom Engineering Advisories and Technical
Notifications.

`reg.x.server.y.registerRetry.baseTimeOut`

For registered line x, set y to the maximum time period the phone waits before trying to re-
register with the server. Used in conjunction with
`reg.x.server.y.registerRetry.maxTimeOut` to determine how long to wait.

60 (default)

10 - 120 seconds

`reg.x.server.y.registerRetry.maxTimeout`

For registered line x, set y to the maximum time period the phone waits before trying to re-
register with the server. Use in conjunction with r
eq x.server.y.registerRetry.baseTimeOut to determine how long to wait. The
algorithm is defined in RFC 5626.

180 - (default)

60 - 1800 seconds
**reg.x.server.y.retryMaxCount**

The number of retries attempted before moving to the next available server.

3 - (default)

0 to 20 - 3 is used when the value is set to 0.

**reg.x.server.y.retryTimeOut**

0 (default) - Use standard RFC 3261 signaling retry behavior.

0 to 65535 - The amount of time (in milliseconds) to wait between retries.

**reg.x.server.y.subscribe.expires**

The phone’s requested subscription period in seconds after which the phone attempts to resubscribe at the beginning of the overlap period.

3600 seconds - (default)

10 - 2147483647 (seconds)

You can use this parameter in conjunction with reg.x.server.y.subscribe.expires.overlap.

**reg.x.server.y.subscribe.expires.overlap**

The number of seconds before the expiration time returned by server x after which the phone attempts to resubscribe. If the server value is less than the configured overlap value, the phone tries to resubscribe at half the expiration time returned by the server.

60 seconds (default)

5 - 65535 seconds

**reg.x.server.y.transport**

The transport method the phone uses to communicate with the SIP server.

DNSnaptr (default) - If reg.x.server.y.address is a hostname and reg.x.server.y.port is 0 or Null, do NAPTR then SRV lookups to try to discover the transport, ports and servers, as per RFC 3263. If reg.x.server.y.address is an IP address, or a port is given, then UDP is used.

TCPpreferred - TCP is the preferred transport; UDP is used if TCP fails.

UDPOnly - Only UDP is used.

TLS - If TLS fails, transport fails. Leave port field empty (defaults to 5061) or set to 5061.

TCPOnly - Only TCP is used.

**reg.x.server.y.useOutboundProxy**

1 (default) - Enables to use the outbound proxy specified in reg.x.outboundProxy.address for server x.

0 - Disable to use the outbound proxy specified in reg.x.outboundProxy.address for server x.
**divert.x.sharedDisabled**

1 (default) - Disables call diversion features on shared lines.
0 - Enables call diversion features on shared lines.
Change causes system to restart or reboot.

**dns.cache.A.x.**

Specify the DNS A address, hostname, and cache time interval.

**dns.cache.A.x.address**

Null (default)
IP version 4 address

**dns.cache.A.x.name**

Null (default)
valid hostname

**dns.cache.A.x.ttl**

The TTL describes the time period the phone uses the configured static cache record. If a
dynamic network request receives no response, this timer begins on first access of the static
record and once the timer expires, the next lookup for that record retries a dynamic network
request before falling back on the static entry and it resets TTL timer again.
300 (default)
300 to 536870912 (2^29), seconds

**dns.cache.NAPTR.x.**

Specify the DNS NAPTR parameters, including: name, order, preference, regexp, replacement,
service, and ttl.

**dns.cache.NAPTR.x.flags**

The flags to control aspects of the rewriting and interpretation of the fields in the record.
Characters are case-sensitive. At this time, only 'S', 'A', 'U', and 'P' are defined as flags. See
**RFC 2915** for details of the permitted flags.
Null (default)
A single character from [A-Z, 0-9]

**dns.cache.NAPTR.x.name**

Null (default)
domain name string - The domain name to which this resource record refers.

**dns.cache.NAPTR.x.order**

0 (default)
0 to 65535 - An integer that specifies the order in which the NAPTR records must be processed to ensure the correct ordering of rules.

**dns.cache.NAPTR.x.preference**

0 (default)

0 to 65535 - A 16-bit unsigned integer that specifies the order in which NAPTR records with equal "order" values should be processed. Low numbers are processed before high numbers.

**dns.cache.NAPTR.x.regexp**

This parameter is currently unused. Applied to the original string held by the client. The substitution expression is applied in order to construct the next domain name to lookup. The grammar of the substitution expression is given in RFC 2915.

Null (default) string containing a substitution expression

**dns.cache.NAPTR.x.replacement**

The next name to query for NAPTR records depending on the value of the flags field. It must be a fully qualified domain-name.

Null (default)

domain name string with SRV prefix

**dns.cache.NAPTR.x.service**

Specifies the service(s) available down this rewrite path. For more information, see RFC 2915.

Null (default)

string

**dns.cache.NAPTR.x.ttl**

The TTL describes the time period the phone uses the configured static cache record. If a dynamic network request receives no response, this timer begins on first access of the static record and once the timer expires, the next lookup for that record retries a dynamic network request before falling back on the static entry and it resets TTL timer again.

300 to 536870912 (2^29), seconds

**dns.cache.A.networkOverride**

0 (default) - Does not allow the static DNS A record entry to take priority over dynamic network DNS.

1 - Allows the static DNS cached A record entry to take priority over dynamic network DNS. Moreover, the DNS TTL value is ignored.

**dns.cache.SRV.x.**

Specify DNS SRV parameters, including: name, port, priority, target, ttl, and weight.
**Network Configuration**

**dns.cache.SRV.x.name**
Null (default)
Domain name string with SRV prefix

**dns.cache.SRV.x.port**
The port on this target host of this service. For more information, see [RFC 2782](https://tools.ietf.org/html/rfc2782).
0 (default)
0 to 65535

**dns.cache.SRV.x.priority**
The priority of this target host. For more information, see [RFC 2782](https://tools.ietf.org/html/rfc2782).
0 (default)
0 to 65535

**dns.cache.SRV.x.target**
Null (default)
domain name string - The domain name of the target host. For more information, see [RFC 2782](https://tools.ietf.org/html/rfc2782).

**dns.cache.SRV.x.ttl**
The TTL describes the time period the phone uses the configured static cache record. If a dynamic network request receives no response, this timer begins on first access of the static record and once the timer expires, the next lookup for that record retries a dynamic network request before falling back on the static entry and it resets TTL timer again.
300 (default)
300 to 536870912 (2^29), seconds

**dns.cache.SRV.x.weight**
A server selection mechanism. For more information, see [RFC 2782](https://tools.ietf.org/html/rfc2782).
0 (default)
0 to 65535

**tcpIpApp.dns.address.overrideDHCP**
Specifies how DNS addresses are set.
0 (default) - DNS address requested from the DHCP server.
1 - DNS primary and secondary address is set using the parameters `tcpIpApp.dns.server` and `tcpIpApp.dns.altServer`.
Change causes system to restart or reboot.
**Note:** If the DHCP server doesn't send the DNS server addresses to the phone, then the values set for the `device.dns.serverAddress` and `device.dns.altSrvAddress` parameters are used. Alternatively, the phone uses the DNS server addresses set using the `tcpIpApp.*` parameters, which override the `device.dns.*` parameters.

**tcpIpApp.dns.domain.overrideDHCP**

Specifies how the domain name is retrieved or set.

0 (default) - Domain name retrieved from the DHCP server, if one is available.

1 - DNS domain name is set using the parameter `tcpIpApp.dns.domain`.

Change causes system to restart or reboot.

**Note:** If the DHCP server doesn't send the DNS domain to the phone, then the value set for `device.dns.domain` is used. Alternatively, the phone uses the DNS domain set using the `tcpIpApp.*` parameter, which overrides the `device.dns.*` parameter.

**dns.cache.dynamicRestore.enable**

1 - Allows the phone to restore the expired cache entries to a specified TTL when the DNS server isn't reachable.

0 (default) - Doesn't allow the phone to restore the expired cache entries to a specified TTL when the DNS server isn't reachable.

**dns.cache.dynamicRestore.ttl**

Specify a TTL value to restore the expired cache entries when the DNS server isn't reachable.

120 (default)

90 to 600 seconds

**reg.x.secureTransportRequiresSrtp**

0 (default) - Doesn't allow the phone to dynamically overwrite the configured values of `reg.x.srtp.offer` parameter and `reg.x.srtp.require` parameter based on the NAPTR response for per line registration.

1 - Allows the phone to dynamically overwrite the configured values of `reg.x.srtp.offer` parameter and `reg.x.srtp.require` parameter based on the NAPTR response for per line registration to enable SRTP only.

**voIpProt.SIP.naptrAllowDuplicateTransport.enable**

0 (Default) - The system ignores NAPTR records with duplicate protocols.

1 - The system considers all NAPTR records, regardless of transport, up to a maximum of 16 records.
Example Static DNS Cache Configuration

The following example shows how to configure static DNS cache using A records IP addresses in SIP server address fields.

The addresses listed in this example are read by UC Software in the order listed.

When the static DNS cache is not used, the `site.cfg` configuration looks as follows:

<table>
<thead>
<tr>
<th>reg.1.address</th>
<th>1001</th>
</tr>
</thead>
<tbody>
<tr>
<td>reg.1.server.1.address</td>
<td>172.23.0.140</td>
</tr>
<tr>
<td>reg.1.server.1.port</td>
<td>5075</td>
</tr>
<tr>
<td>reg.1.server.1.transport</td>
<td>UDPMOnly</td>
</tr>
<tr>
<td>reg.1.server.2.address</td>
<td>172.23.0.150</td>
</tr>
<tr>
<td>reg.1.server.2.port</td>
<td>5075</td>
</tr>
<tr>
<td>reg.1.server.2.transport</td>
<td>UDPMOnly</td>
</tr>
</tbody>
</table>

When the static DNS cache is used, the `site.cfg` configuration looks as follows:

<table>
<thead>
<tr>
<th>reg.1.address</th>
<th>1001</th>
</tr>
</thead>
<tbody>
<tr>
<td>reg.1.server.1.address</td>
<td>sips server.example.com</td>
</tr>
<tr>
<td>reg.1.server.1.port</td>
<td>5075</td>
</tr>
<tr>
<td>reg.1.server.1.transport</td>
<td>UDPMOnly</td>
</tr>
<tr>
<td>reg.1.server.2.address</td>
<td>sips server.example.com</td>
</tr>
<tr>
<td>reg.1.server.2.port</td>
<td>5075</td>
</tr>
<tr>
<td>reg.1.server.2.transport</td>
<td>UDPMOnly</td>
</tr>
<tr>
<td>dns.cache.A.1.name</td>
<td>sips server.example.com</td>
</tr>
<tr>
<td>dns.cache.A.1.ttl</td>
<td>3600</td>
</tr>
<tr>
<td>dns.cache.A.1.address</td>
<td>172.23.0.140</td>
</tr>
<tr>
<td>dns.cache.A.2.name</td>
<td>sips server.example.com</td>
</tr>
<tr>
<td>dns.cache.A.2.ttl</td>
<td>3600</td>
</tr>
<tr>
<td>dns.cache.A.2.address</td>
<td>172.23.0.150</td>
</tr>
</tbody>
</table>

Example: Static DNS Cache with A Records

This example shows how to configure static DNS cache where your DNS provides A records for `reg.x server.x address` but not SRV. In this case, the static DNS cache on the phone provides SRV records. For more information, see RFC 3263.

When the static DNS cache is not used, the `site.cfg` configuration looks as follows:

<table>
<thead>
<tr>
<th>reg.1.address</th>
<th>1000@sips server.example.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>reg.1.server.1.address</td>
<td>primary.sips server.example.com</td>
</tr>
<tr>
<td>reg.1.server.1.port</td>
<td>5075</td>
</tr>
<tr>
<td>reg.1.server.1.transport</td>
<td>UDPMOnly</td>
</tr>
<tr>
<td>reg.1.server.2.address</td>
<td>secondary.sips server.example.com</td>
</tr>
<tr>
<td>reg.1.server.2.port</td>
<td>5075</td>
</tr>
<tr>
<td>reg.1.server.2.transport</td>
<td>UDPMOnly</td>
</tr>
</tbody>
</table>

When the static DNS cache is used, the `site.cfg` configuration looks as follows:
The `reg.1.server.1.port` and `reg.1.server.2.port` values in this example are set to null to force SRV lookups.

### Example: Static DNS Cache with NAPTR and SRV Records

This example shows how to configure static DNS cache where your DNS provides NAPTR and SRV records for `reg.x.server.x.address`.

When the static DNS cache is not used, the `site.cfg` configuration looks as follows:

```
+----+----------------+----------------+---------------------+
|    | reg.1.address  | reg.1.server.1.address | reg.1.server.1.port |
|----+----------------+----------------+---------------------+
|    | 1002           | sipserver.example.com | 5075                |
|----+----------------+----------------+---------------------+
```

When the static DNS cache is used, the `site.cfg` configuration looks as follows:

```
+----+----------------+----------------+---------------------+
|    | reg.1.address  | reg.1.server.1.address | reg.1.server.1.port |
|----+----------------+----------------+---------------------+
|    | 1002           | sipserver.example.com | 5075                |
|----+----------------+----------------+---------------------+
```

Note: The `reg.1.server.1.port` and `reg.1.server.2.port` values in this example are set to null to force SRV lookups.
**IP Type-of-Service**

The type-of-service field in an IP packet header consists of four type-of-service (TOS) bits and a 3-bit precedence field.

Each TOS bit can be set to either 0 or 1. The precedence field can be set to a value from 0 through 7. The type of service can be configured specifically for RTP packets and call control packets, such as SIP signaling packets.

**IP Type-of-Service Parameters**

You can configure the IP TOS feature specifically for RTP and call control packets, such as SIP signaling packets.

Type of Service (ToS) and the Differentiated Services Code Point (DSCP) allows specification of a datagrams desired priority and routing through low-delay, high-throughput, or highly-reliable networks.

The IP ToS header consists of four ToS bits and a 3-bit precedence field. DSCP replaces the older ToS specification and uses a 6-bit DSCP in the 8-bit differentiated services field (DS field) in the IP header.

The parameters listed below configure the type of service field RTP and call control packets for Quality of Service (QoS).

```markdown
qos.ethernet.tcpQosEnabled
```

0 (default) - The phone does not send configured QoS priorities for SIP over TCP transport.
The phone sends configured QoS priorities for SIP over TCP transport.
Change causes system to restart or reboot.

**qos.ip.callControl.dscp**
Specify the DSCP of packets.
If the value is set to the default NULL the phone uses `qos.ip.callControl.*` parameters.
If the value is not NULL, this parameter overrides `qos.ip.callControl.*` parameters.
Change causes system to restart or reboot.

**qos.ip.callControl.max_reliability**
Set the max reliability bit in the IP ToS field of the IP header used for call control.
0 (default) - The bit in the IP ToS field of the IP header is not set.
1 - The bit is set.
Change causes system to restart or reboot.

**qos.ip.callControl.max_throughput**
Set the throughput bit in the IP ToS field of the IP header used for call control.
0 (default) - The bit in the IP ToS field of the IP header is not set.
1 - The bit is set.
Change causes system to restart or reboot.

**qos.ip.callControl.min_cost**
Set the min cost bit in the IP ToS field of the IP header used for call control.
0 (default) - The bit in the IP ToS field of the IP header is not set.
1 - The bit is set.
Change causes system to restart or reboot.

**qos.ip.callControl.min_delay**
Set the min delay bit in the IP ToS field of the IP header used for call control.
1 (default) - The bit is set.
0 - The bit in the IP ToS field of the IP header is not set.
Change causes system to restart or reboot.

**qos.ip.callControl.precedence**
Set the min delay bit in the IP ToS field of the IP header used for call control.
5 (default)
0 - 7
Change causes system to restart or reboot.
qos.ip.rtp.dscp
Specify the DSCP of packets.
If the value is set to the default NULL, the phone uses quality.ip.rtp.* parameters.
If the value is not NULL, this parameter overrides quality.ip.rtp.* parameters.
- Null (default)
- 0 to 63
- EF
- Any of AF11, AF12, AF13, AF21, AF22, AF23, AF31, AF32, AF33, AF41, AF42, AF43

Change causes system to restart or reboot.

qos.ip.rtp.max_reliability
Set the max reliability bit in the IP ToS field of the IP header used for RTP.
0 (default) - The bit in the IP ToS field of the IP header is not set.
1 - The bit is set.
Change causes system to restart or reboot.

qos.ip.rtp.max_throughput
Set the throughput bit in the IP ToS field of the IP header used for RTP.
0 (default) - The bit in the IP ToS field of the IP header is not set.
1 - The bit is set.
Change causes system to restart or reboot.

qos.ip.rtp.min_cost
Set the min cost bit in the IP ToS field of the IP header used for RTP.
0 (default) - The bit in the IP ToS field of the IP header is not set.
1 - The bit is set.
Change causes system to restart or reboot.

qos.ip.rtp.min_delay
Set the min delay bit in the IP ToS field of the IP header used for RTP.
1 (default) - The bit is set.
0 - The bit in the IP ToS field of the IP header is not set.
Change causes system to restart or reboot.

qos.ip.rtp.precedence
Set the precedence bit in the IP ToS field of the IP header used for RTP.
5 (default)
0 - 7
Change causes system to restart or reboot.

**qos.ip.rtp.video.dscp**

Allows you to specify the DSCP of packets.

If the value is set to the default NULL, the phone uses `qos.ip.rtp.video.*` parameters.

If the value is not NULL, this parameter overrides `qos.ip.rtp.video.*` parameters.

- NULL (default)
- 0 to 63
- EF
- Any of AF11, AF12, AF13, AF21, AF22, AF23, AF31, AF32, AF33, AF41, AF42, AF43

Change causes system to restart or reboot.

**qos.ip.rtp.video.max_reliability**

Set the reliability bits in the IP ToS field of the IP header used for RTP video.

- 0 (default) - The bit in the IP ToS field of the IP header is not set.
- 1 - The bit is set.

Change causes system to restart or reboot.

**qos.ip.rtp.video.max_throughput**

Set the throughput bits in the IP ToS field of the IP header used for RTP video.

- 0 (default) - The bit in the IP ToS field of the IP header is not set.
- 1 - The bit is set.

Change causes system to restart or reboot.

**qos.ip.rtp.video.min_cost**

Set the min cost bits in the IP ToS field of the IP header used for RTP video.

- 0 (default) - The bit in the IP ToS field of the IP header is not set.
- 1 - The bit is set.

Change causes system to restart or reboot.

**qos.ip.rtp.video.min_delay**

Set the min delay bits in the IP ToS field of the IP header used for RTP video.

- 1 (default) - The bit is set.
- 0 - The bit in the IP ToS field of the IP header is not set.

Change causes system to restart or reboot.

**qos.ip.rtp.video.precedence**

Set the precedence bits in the IP ToS field of the IP header used for RTP video.
SIP Instance Support

In environments where multiple phones are registered using the same address of record (AOR), the phones are identified by their IP address.

However, firewalls set up in these environments can regularly change the IP addresses of phones for security purposes. You can configure SIP instance to identify individual phones instead of using IP addresses. This feature complies with RFC 3840.

SIP Instance Parameter

The parameter `reg.x.gruu` provides a contact address to a specific user agent (UA) instance, which helps to route the request to the UA instance and is required in cases in which the REFER request must be routed to the correct UA instance. Refer to the following list for the parameters to configure this feature.

`reg.x.gruu`

1 - The phone sends sip.instance in the REGISTER request.
0 (default) - The phone does not send sip.instance in the REGISTER request.

Provisional Polling of Phones

You can configure phones to poll the server for provisioning updates automatically, and you can set the phone's automatic provisioning behavior to one of the following:

- **Absolute**—The phone polls at the same time every day.
- **Relative**—The phone polls every `x` seconds, where `x` is a number greater than 3600.
- **Random**—The phone polls randomly based on a set time interval.
  - If the time period is less than or equal to one day, the first poll is at a random time between when the phone starts up and the polling period. Afterward, the phone polls every `x` seconds.
  - If you set the polling period to be greater than one day with the period rounded up to the nearest day, the phone polls on a random day based on the phone's MAC address and within a random time set by the start and end polling time.

Provisional Polling Parameters

Use the parameters in the following list to configure provisional polling.

Note: If `prov.startupCheck.enabled` is set to 0, then the phones do not look for the sip.id or the configuration files when they reboot, lose power, or restart. Instead, they look only when receiving a checksync message, a polling trigger, or a manually started update from the menu or web UI.
Some files such as bitmaps, .wav, the local directory, and any custom ringtones are downloaded each time as they are stored in RAM and lost with every reboot.

**prov.polling**
To enable polling and set the mode, period, time, and time end parameters.

**prov.polling.enabled**

0 (default) - Disables the automatic polling for upgrades.
1 - Initiates the automatic polling for upgrades.

**prov.polling.mode**
The polling modes for the provisioning server.

- **abs** (default) - The phone polls every day at the time specified by `prov.polling.time`.
- **rel** - The phone polls after the number of seconds specified by `prov.polling.period`.
- **random** - The phone polls at random between a starting time set in `prov.polling.time` and an end time set in `prov.polling.timeRandomEnd`.

If you set the polling period in `prov.polling.period` to a time greater than 86400 seconds (one day) polling occurs on a random day within that polling period and only between the start and end times. The day within the period is decided based upon the phone's MAC address and does not change with a reboot whereas the time within the start and end is calculated again with every reboot.

**prov.polling.period**
The polling period is calculated in seconds and is rounded up to the nearest number of days in an absolute and random mode. If this is set to a time greater than 86400 (one day) polling occurs on a random day based on the phone's MAC address.

86400 (default) - Number of seconds in a day.
Integer - An integer value greater than 3600 seconds.

**prov.polling.time**
The start time for polling on the provisioning server.
03:00 (default)

**prov.polling.timeRandomEnd**
The stop time for polling on the provisioning server.
Null (default)

**Example Provisional Polling Configuration**
The following are examples of polling configurations you can set up:
• If `prov.polling.mode` is set to `rel` and `prov.polling.period` is set to 7200, the phone polls every two hours.
• If `prov.polling.mode` is set to `abs` and `prov.polling.timeRandomEnd` is set to 04:00, the phone polls at 4am every day.
• If `prov.polling.mode` is set to `random`, `prov.polling.period` is set to 604800 (7 days), `prov.polling.time` is set to 01:00, `prov.polling.timeRandomEnd` is set to 05:00, and you have 25 phones, a random subset of those 25 phones, as determined by the MAC address, polls randomly between 1am and 5am every day.
• If `prov.polling.mode` is set to `abs` and `prov.polling.period` is set to 2328000, the phone polls every 20 days.

SIP Subscription Timers

You can configure a subscription expiry independently of the registration expiry.

You can also configure an overlap period for a subscription independently of the overlap period for the registration, and a subscription expiry and subscription overlap for global SIP servers and per-registration SIP servers. Note that per-registration configuration parameters override global parameters. If you have not explicitly configured values for any user features, the default subscription values are used.

SIP Subscription Timers Parameters

Use the parameters in the following list to configure when a SIP subscription expires and when expiration dates overlap.

`voIpProt.server.x.subscribe.expires`

The phone’s requested subscription period in seconds after which the phone attempts to resubscribe at the beginning of the overlap period.

3600 - (default)
10 - 2147483647

`voIpProt.server.x.subscribe.expires.overlap`

The number of seconds before the expiration time returned by server x after which the phone attempts to resubscribe. If the server value is less than the configured overlap value, the phone tries to resubscribe at half the expiration time returned by the server.

60 - (default)
5 - 65535 seconds

`reg.x.server.y.subscribe.expires`

The phone’s requested subscription period in seconds after which the phone attempts to resubscribe at the beginning of the overlap period.

3600 seconds - (default)
10 - 2147483647 (seconds)

You can use this parameter in conjunction with `reg.x.server.y.subscribe.expires.overlap`.  

374
**reg.x.server.y.subscribe.expires.overlap**

The number of seconds before the expiration time returned by server x after which the phone attempts to resubscribe. If the server value is less than the configured overlap value, the phone tries to resubscribe at half the expiration time returned by the server.

60 seconds (default)

5 - 65535 seconds

---

**Incoming Network Signaling Validation**

You can choose from the following optional levels of security for validating incoming network signaling:

- Source IP address validation
- Digest authentication
- Source IP address validation and digest authentication

**Network Signaling Validation Parameters**

The following list includes the parameters you can use to specify the validation type, method, and the events for validating incoming network signaling.

**voIpProt.SIP.requestValidation.x.method**

- Null (default) - No validation is made.
- Source - Ensure request is received from an IP address of a server belonging to the set of target registration servers.
- digest - Challenge requests with digest authentication using the local credentials for the associated registration (line).
- both or all - Apply both of the above methods.

Change causes system to restart or reboot.

**voIpProt.SIP.requestValidation.x.request**

Sets the name of the method for which validation will be applied.

Null (default)

INVITE, ACK, BYE, REGISTER, CANCEL, OPTIONS, INFO, MESSAGE, SUBSCRIBE, NOTIFY, REFER, PRACK, UPDATE

**Note:** Intensive request validation may have a negative performance impact due to the additional signaling required in some cases.

Change causes system to restart or reboot.

**voIpProt.SIP.requestValidation.x.request.y.event**

Determines which events specified with the Event header should be validated; only applicable when voIpProt.SIP.requestValidation.x.request is set to SUBSCRIBE or NOTIFY.

Null (default) - all events will be validated.
A valid string - specified event will be validated.
Change causes system to restart or reboot.

Session Traversal Utilities for NAT (STUN)
Poly UC Software supports Session Traversal Utilities for NAT (STUN), a network protocol used in NAT traversal for real-time IP communications, such as voice, video, and messaging.
You can configure the phone to act as a STUN client to send request to STUN Server to discover the public IP and port(s). You can also configure the phone to send keep-alive messages to refresh NAT bindings.

STUN Parameters
This section lists parameters that configure Simple Traversal of UDP though NAT (STUN).

**feature.nat.stun.enabled**
0 (default) - Disable the STUN.
1 - Enable the STUN. SIP responses are sent to the source IP address and source port where the request originated. If you also enable parameter `voIpProt.SIP.rport`, then the phone adds the received IP address and port in the VIA header while generating the response.
Change causes system to restart or reboot.

**nat.stun.server**
Enter a STUN server IP address or domain name.
Null (default)
Change causes system to restart or reboot.

**nat.stun.port**
Set the STUN server port number.
3478 (default)
1 to 65535
Change causes system to restart or reboot.

**reg.x.nat.traversal.mode**
Enable or disable NAT traversal mode with STUN for signaling and media on the basis of the phone-level STUN feature.
Auto (default) - Apply NAT configuration to both media and signaling per registration.
Disabled - The phone doesn’t use STUN for NAT traversal for this registration.
For example, if `feature.nat.stun.enabled` is set to 1, and `reg.x.nat.traversal.mode` is set as Auto, the STUN feature is enabled for signaling and media for the registered line.
**nat.refresh.interval**

Set the time interval for the phone to send STUN binding indications to keep the NAT port open and the phone reachable.

- **30 seconds (default)** - The phone sends STUN binding indications for every 30 seconds to keep the NAT port open and the phone reachable.
- **0 second** - Disable STUN Binding indication to refresh NAT bindings.
- **3600 seconds**

**nat.device.pollInterval**

Set the time interval for the phone to send STUN binding request to the STUN server to detect whether NAT device is rebooted.

- **120 seconds (default)** - The phone sends the STUN binding requests to the STUN server for every 120 seconds. If NAT IP address or the port details in the STUN binding response don’t match with the previous binding response, the phone automatically restarts.
- **0** - The phone doesn’t check whether NAT device is rebooted. If NAT device is rebooted and the NAT IP address or the port is changed, the phone doesn’t receive any incoming messages as the IP address and port details published in SIP register message don’t match. You need to restart the phone manually to make the changes effective. Poly recommends not to set the value as 0 second.
- **900 seconds**

**GZIP Encoding of SIP INFO Messages**

To reduce bandwidth, the phone sends notifications to the server in gzip format.

**GZIP Encoding Parameter**

Use the following parameter to configure GZIP Encoding to send notifications to the server.

**voIpProt.SIP.gzipEncoding.enable**

- **Enable or disable GZIP encoding.**
- **0** - Disabled (Default)
- **1** - Enabled

**DHCP IP Address**

The phone enables DHCP by default.

If the phone can’t communicate with the DHCP server on startup, the phone’s status bar reports “Network Down.”

The phone communicates with the DHCP server every 5 minutes to acquire an IP Address or for lease renewal.
DHCP IP Address Cache

Poly UC Software supports Dynamic Host Configuration Protocol (DHCP) IP address cache to retain IP addresses on the phones when the DHCP server becomes unavailable.

When you enable the IP address cache feature, there isn’t a service interruption even if the IP address lease time expires and the DHCP server doesn’t respond. The phone periodically attempts to resume DHCP service with a new DHCP Discover message for the entire time the cached IP address is in use.

DHCP IP address cache stores the following lease parameters:

- Interface
- IP Address
- Subnet Mask
- Gateway
- DNS Server
- Domain Name

DHCP IP address cache has the following limitations:

**Important:** If a DHCP server restarts and loses lease details, enabling DHCP IP address cache can lead to IP address conflicts on the phones. This results in cascading service outages.

- The phones don’t cache DHCP option 99 values for Enhanced 911 location services. If you enable IP address cache, emergency calling services may be affected in case of WAN outage only.
- If a DHCP server restarts and loses lease details, enabling DHCP IP address cache can lead to IP address conflicts on the phones. This results in cascading service outages.
- DHCP IP address cache supports only IPv4 addresses. DHCP IP address cache doesn’t support IPv6 addresses at this time.
- DHCP IP address cache doesn’t support DHCP VLAN Discovery (DVD).
- If you move a phone from one VLAN to another VLAN where DHCP doesn’t respond, the phone continues to use the cached IP address.
- The phones can’t update the software using DHCP IP address cache. When the phones attempt to update the Polycom UC software without DHCP server availability, the phones experience a reboot loop. This continuous reboot loop occurs only when:
  - A cached IP address is in use.
  - The DHCP server is unavailable.
  - A software provisioning server is available.
  - New software is available on the provisioning server.
- You can use DHCP IP address cache only for the UC Software application; you cannot use it for the Updater.

**DHCP IP Address Cache Configuration Parameters**

Use the following parameters to configure DHCP IP address cache.
device.net.cachedIPAddress
0 (default) – IP addresses isn't cached.
1- If a DHCP response isn't received, the phone uses the last assigned IP address, provided one is cached already. A DHCP discover message is retried every device.net.cachedIPAddressRetryTime second.

device.net.cachedIPAddressRetryTime
Specify the time in seconds to send new DHCP to discover messages when using a cached IP address.

Note: This is only applicable when device.net.cachedIPAddress is enabled.

3600 (default)
300 - 7200

Wireless Network Connectivity (Wi-Fi)
The Poly CCX 600 and 700 business media phones support several wireless modes, security options, and radio controls. CCX phones don’t support connecting to wireless networks that require web authentication.

To ensure the best performance in your location, set a proper country code with the parameter device.wifi.country before enabling Wi-Fi.

Note: If device.wifi.country is not set, the phone will operate in a world safe mode restricting Wi-Fi channels and power.

You can configure Wi-Fi options to display in the phone’s basic settings menu to allow users to manually add a Wi-Fi network. You can also configure the phone to display the Wi-Fi icon on the phone’s status bar and home screen.

Enabling Wi-Fi automatically disables the Ethernet port. You can’t use Wi-Fi and Ethernet simultaneously to connect phones to your network. When you connect the system to your network over Wi-Fi, only audio-only calls are available. The phones don’t support Wi-Fi captive portals or Wireless Display (WiDi).

Note: When you provision via Wi-Fi connection to the network, the phone looks for files on the provisioning server using the LAN MAC address and not the Wi-Fi MAC address.

Enable Wi-Fi
You can wirelessly connect phones to your network using Wi-Fi, which is disabled by default.

When you enable Wi-Fi, the system reboots.

Procedure
1. Go to Settings > Advanced > Administration Settings > Network Configuration > Network Interfaces > Wi-Fi Menu, and turn Wi-Fi to On.
The phone reboots.

2. After the phone restarts, go to Settings > Advanced > Administration Settings > Network Configuration > Network Interfaces > Wi-Fi Menu to view available networks.

3. Select a network and press Connect.

**Configure Wireless Network Settings**

You can manually configure the phone to connect to a wireless network by selecting an enterprise-based network and EAP method for better security.

**Procedure**

1. Go to Settings > Advanced > Administrator Settings > Network Configuration > Network Interfaces > Wi-Fi Menu.

2. Turn Wi-Fi to On.

   The phone reboots.

3. After the phone reboots, go to Settings > Advanced > Administration Settings > Network Configuration > Network Interfaces > Wi-Fi Menu to view available networks.

4. Configure the following wireless network settings:
   
   a. Select the SSID name of the wireless network.
   
   b. Select the security type of the wireless network.
   
   c. Optional: If you have an enterprise-based network, enter the User ID and Password.
   
   d. Select one of the following EAP-Method types for authentication:
      
      • EAP-PEAP-MSCHAPv2
      
      • EAP-TTLS-MSCHAPv2
      
      • EAP-PWD
      
      • EAP-TLS

5. Select a network and press Connect.

**Wi-Fi Parameters**

The parameters you configure depend on the security mode of your organization and whether or not you enable DHCP.

CCX 600 and 700 business media phones include a security-restrictive worldwide Wi-Fi country code setting.

The CCX 600 and 700 business media phones support the following Wi-Fi security modes:

   • WPA PSK
   
   • WPA2 PSK
   
   • WPA2 Enterprise

**device.wifi.country**

   NULL (default)
   
   Two-letter country code
**device.wifi.dhcpEnabled**

Enable or disable DHCP for Wi-Fi.

- 0 (default) - Disable
- 1 - Enable

**device.wifi.enabled**

Enable or disable Wi-Fi.

- 0 (default) - Disable
- 1 - Enable

**device.wifi.ipAddress**

Enter the IP address of the wireless device if you are not using DHCP.

- 0.0.0.0 (default)
- String

**device.wifi.ipGateway**

Enter the IP gateway address for the wireless interface if not using DHCP.

- 0.0.0.0 (default)
- String

**device.wifi.psk.key**

Enter the hexadecimal key or ASCII passphrase.

- 0xFF (default)
- String

**device.wifi.psk.keyType**

Set the Pre-Shared Key (PSK) type.

- 0 (default) - Passphrase.
- 1 - Hexadecimal key.

**device.wifi.securityMode**

Specify the wireless security mode.

- NULL (default)
- WPA-PSK
- WPA2-PSK
- WPA2-Enterprise
device.wifi.ssid
Set the Service Set Identifier (SSID) of the wireless network.
SSID1 (default)
SSID

device.wifi.subnetMask
Set the network mask address of the wireless device if not using DHCP.
255.0.0.0 (default)
String

device.wifi.wpa2Ent.method
Set the Extensible Authentication Protocol (EAP) to use for 802.1X authentication.
Null (default)
EAP-PEAPv0/MSCHAPv2
EAP-TTLS-MSCHAPv2
EAP-PEAPv0-NONE
EAP-TTLS-NONE
EAP-PWD
EAP-TLS

device.wifi.wpa2Ent.password
The WPA2-Enterprise password.

device.wifi.wpa2Ent.user
The WPA2-Enterprise user name.

feature.wifi.UserSettings.enabled
1 (default) – Display Wi-Fi menu options on the phone menu.
0 – Wi-Fi menu options do not display on the phone menu.

homeScreen.wifi.enable
1 (default) – Display the Wi-Fi icon on the phone’s Home screen.
0 – Do not display the Wi-Fi icon on the phone’s Home screen.

status.wifi.icon.enable
1 (default) – Display the Wi-Fi icon on the status bar of the phone’s screen. Users can access Wi-Fi settings by selecting the Wi-Fi icon.
0 – Do not display the Wi-Fi icon on the status bar.
Configuring Bluetooth

You can enable Bluetooth to allow users to connect and pair compatible Bluetooth devices such as a mobile phone, tablet, laptop, or headset with Poly CCX.

Bluetooth is disabled by default. You must configure the following parameters to enable Bluetooth and allow devices to find and pair with your CCX:

- `feature.bluetooth.enabled`
- `bluetooth.radioOn`

After you enable Bluetooth, you can pair and cache a maximum of six Bluetooth devices with the CCX phone and connect a maximum of two devices at a time, or pair the Poly CCX phone with one of your devices.

When you pair your Bluetooth headset to the phone, you can use your headset to manage call audio. When you pair the phone to your PC, tablet, or mobile phone, you can use the phone to answer a call, end a call, or reject a call.

Note that using a Bluetooth headset can affect voice quality on the phone due to inherent limitations with Bluetooth technology. You may not experience the highest voice quality when using a Bluetooth headset while the 2.4 GHz band is enabled or while you are in an environment with many other Bluetooth devices.

**Note:** Bluetooth is supported with Poly CCX 500, and 600 business media phones only. This feature is not supported on Poly CCX 400 business media phones.

**Bluetooth Parameters**

Use the parameters in the following list to configure Bluetooth.

**bluetooth.device.name**

- NULL (default)
- UTF-8 string

  Enter the name of the device that broadcasts over Bluetooth to other devices.

**bluetooth.discoverableTimeout**

  Set the time in seconds after which other devices can discover this device over Bluetooth.

  - 0 (default) - Other devices can always discover this device over Bluetooth.
  - 0 - 3600 seconds

**bluetooth.pairedDeviceMemorySize**

  Set the maximum number of devices that can be paired and cached as paired on the phone.

  - 10 (default)
  - 0 - 10
**bluetooth.radioOn**

0 - The Bluetooth radio transmitter/receiver is off.
1 (default) - The Bluetooth radio is on. You must turn on the Bluetooth radio to allow devices to connect over Bluetooth.

**feature.bluetooth.enabled**

For high security environments.
1 (default) – Enable Bluetooth and the Bluetooth phone screen icon.
0 - Disable Bluetooth and the Bluetooth phone screen icon.

**Supported Bluetooth Profiles**

To pair devices via Bluetooth, you must ensure that paired devices support required Bluetooth profiles. Poly CCX phones support the following Bluetooth application profiles:

- Headset Profile (HSP) (both Gateway and Device)
- Hands-Free Profile (HFP) (both Gateway and Device)
- Advanced Audio Distribution Profile (A2DP) (Both Source and Sink)
Third-Party Servers

Topics:

- Microsoft Exchange Integration
- Ribbon Communications Server
- BroadSoft BroadWorks Server
- Configuring uaCSTA

This section explains certain features you can configure with third-party servers.

**Microsoft Exchange Integration**

After you connect phones with the Exchange Server, you can do the following:

- Verify the status of Exchange Server services on each phone
- View the status of each service in the system web interface

**Integrating with Microsoft Exchange**

You can integrate with Microsoft Exchange using one of the following methods:

- Exchange Server auto-discover
- Provision the phone with the Microsoft Exchange address
- System web interface

**Provision the Microsoft Exchange Calendar**

You can provision your phones with the Microsoft Exchange calendar.

**Procedure**

1. Add the following parameters to one of your configuration files:
   - `feature.exchangeCalendar.enabled=1`
   - `exchange.server.url=https://<example URL>`

**Enable Microsoft Exchange Calendar Using the System Web Interface**

You can use the system web interface to manually enable your phones with the Microsoft Exchange calendar. This option is useful for troubleshooting faulty auto-discovery.

You can enable the Microsoft Exchange calendar through the system web interface for only one phone at a time.

**Procedure**

1. Log in to the system web interface using admin credentials (default password 456).
2. Go to **Settings > Applications > Exchange Applications**.
3. In the Exchange Calendar field, select Enable.
4. Enter the exchange web services URL using a Microsoft Exchange Server URL.
   For example https://<mail.com>/ews/exchange.asmx.
5. Select Save.
6. Select Yes.
   The Calendar icon displays on your phone screen.

Verify the Microsoft Exchange Integration
Verify that all of the Exchange services work properly.

Procedure
   » Do one of the following:
      • On the phone’s local interface, go to Settings > Status > Diagnostics > Warnings.
      • On the system web interface, go to Diagnostics > Skype for Business Status > Exchange Client.

Configuring the Microsoft Exchange Server
You can configure the following settings to use Microsoft Exchange services on your phones.

Visual Voicemail
On the Exchange Server, enable unified messaging and enable messages to play on the phone for each user.

Calendar Month View
On the exchange server, you can enable the month view option for users to retrieve the calendar events for all the days in the month.
The Month View option is disabled by default.

Calendar Month View Parameters
The following parameters configure the month view.

calendar.monthView.enabled
   0 (default) - Disables the Month View soft key.
   1 - Enables the Month View soft key.
Synchronizing Call Logs

On the Exchange Server, you can enable the option to save calls logs to each user’s conversation history in Outlook.

Call Log Synchronization Parameter

Use the following parameter to configure call logs.

feature.exchangeCallLog.enabled

1 (default) - The Exchange call log feature is enabled and the user call log history of Missed, Received, and outgoing calls can be retrieved on the phone.

You must also enable the parameter feature.exchangeCalendar.enabled to use the Exchange call log feature. If you disable feature.exchangeCalendar.enabled, also disable feature.exchangeCallLog.enabled to ensure call log functionality.

0 (default) - The Exchange call log feature is disabled and the user call logs history cannot be retrieved from the Exchange server.

Microsoft Exchange Parameters

The following parameters configure Microsoft Exchange integration.

exchange.meeting.alert.followOfficeHours

1 (default) - Enable audible calendar alerts during business hours.
0 - Disable audible calendar alerts.

exchange.meeting.alert.tonePattern

 positiveConfirm (default) - Set the tone pattern of the reminder alerts using any tone specified by se.pat.*.

exchange.meeting.alert.toneVolume

10 (default) - Set the volume level of reminder alert tones.
0 - 17

exchange.meeting.allowScrollingToPast

0 (default) - Do not allow scrolling up in the Day calendar view to see recently past meetings.
1 - Allow scrolling up in the Day calendar view to see recently past meetings.

exchange.meeting.parseOption

Select a meeting invite field to fetch a VMR or meeting number from.
Location (default)
All
LocationAndSubject
Description
Change causes a reboot.

**exchange.meeting.phonePattern**

NULL (default)

string

The pattern used to identify phone numbers in meeting descriptions, where "x" is a digit or an asterisk(*) and "|" separates alternative patterns (for example, xxx-xxx-xxxx|604.xxx.xxxx).

**exchange.meeting.realConnectProcessing.outboundRegistration**

Choose a line number to use to make calls on Polycom RealConnect technology.

2 (default)

1 - 34

Change causes system to restart or reboot.

**exchange.meeting.realConnectProcessing.prefix.domain**

Define the One-Touch Dial meeting invite prefix domain. Example: "mypolycom.com"

**exchange.meeting.realConnectProcessing.prefix.value**

Define the One-Touch Dial meeting invite prefix value.

**exchange.meeting.realConnectProcessing.skype.enabled**

0 (default) – Disable the Skype for Business meeting on Polycom RealConnect technology. 1 - Enable the Skype for Business meeting on Polycom RealConnect technology.

Change causes system to restart or reboot.

**exchange.meeting.reminderEnabled**

1 (default) - Meeting reminders are enabled. 0 - Meeting reminders are disabled.

**exchange.meeting.reminderInterval**

300 seconds (default)

60 - 900 seconds

Set the interval at which phones display reminder messages.

**exchange.meeting.reminderSound.enabled**

1 (default) - The phone makes an alert sound when users receive reminder notifications of calendar events. Note that when enabled, alert sounds take effect only if exchange.meeting.reminderEnabled is also enabled.

0 - The phone does not make an alert sound when users receive reminder notifications of calendar events.
**exchange.meeting.reminderType**

Customize the calendar reminder and tone.

- 2 (default) - The reminder is always audible and visual.
- 1 - The first reminder is audible and visual reminders are silent.
- 0 - All reminders are silent.

**exchange.meeting.reminderWake.enabled**

- 1 (default) - The phone wakes from low power mode after receiving a calendar notification.
- 0 - The phone stays in low power mode after receiving a calendar notification.

**exchange.pollInterval**

The interval, in milliseconds, to poll the Exchange server for new meetings.

- 30000 (default)
- 4000 minimum
- 60000 maximum

**exchange.server.url**

- NULL (default)
- string

The Microsoft Exchange server address.

**feature.EWSAutodiscover.enabled**

If you configure `exchange.server.url` and set this parameter to 1, preference is given to the value of `exchange.server.url`.

Generic Base Profile default is 0.

- 1 - Exchange autodiscovery is enabled and the phone automatically discovers the Exchange server using the email address or SIP URI information.
- 0 - Exchange autodiscovery is disabled on the phone and you must manually configure the Exchange server address.

**feature.exchangeCalendar.enabled**

Generic Base Profile default is 0.

- 0 - The calendaring feature is disabled.
- 1 - The calendaring feature is enabled.

You must enable this parameter if you also enable `feature.exchangeCallLog.enabled`. If you disable `feature.exchangeCalendar.enabled`, also disable `feature.exchangeCallLog.enabled` to ensure call log functionality.
**exchange.multipleCalendarEvents.enabled**

1 (default) - Multiple calendar events display if at least two events begin within 15 minutes of each other.

0 - Only the next calendar event displays.

**feature.exchangeContacts.enabled**

Generic Base Profile default is 0.

1 - The Exchange call log feature is enabled and users can retrieve the call log histories for missed, received, and outgoing calls.

0 - The Exchange call log feature is disabled and users cannot retrieve call logs histories.

You must also enable the parameter `feature.exchangeCallLog.enabled` to use the Exchange call log feature.

**feature.exchangeVoiceMail.enabled**

Generic Base Profile default is 0.

1 - The Exchange voicemail feature is enabled and users can retrieve voicemails stored on the Exchange server from the phone.

0 - The Exchange voicemail feature is disabled and users cannot retrieve voicemails from Exchange Server on the phone.

You must also enable `feature.exchangeCalendar.enabled` to use the Exchange contact feature.

**feature.exchangeVoiceMail.skipPin.enabled**

0 (default) - Enable PIN authentication for Exchange Voicemail. Users are required to enter their PIN before accessing Exchange Voicemail.

1 - Disable PIN authentication for Exchange Voicemail. Users are not required to enter their PIN before accessing Exchange Voicemail.

**feature.exchange2019.interop.enabled**

0 (default) - Disabled

1 - The device sends a read notification for voicemail after playing to mark the voicemail has been read on the server.

**feature.lync.abs.enabled**

Generic Base Profile default is 0.

1 - Enable comprehensive contact search in the Skype for Business address book service.

0 - Disable comprehensive contact search in the Skype for Business address book service.

**feature.lync.abs.maxResult**

Define the maximum number of contacts to display in a Skype for Business address book service contact search.

12 (default)
feature.wad.enabled
Do not disable this parameter if you are using Skype Online or Web Sign-In.
1 (default) – The phone attempts to use Web auto-discovery and if no FQDN is available, falls back to DNS.
0 - The phone uses DNS to locate the server FQDN and does not use Web auto-discovery. Do not disable this parameter when using Skype for Business Online and Web Sign In.

feature.contacts.readonly
0 (default) - Skype for Business Contacts are editable.
1 - Skype for Business are read-only.

up.oneTouchDirectory
Generic Base Profile default is 0.
1 - The Skype for Business Search icon displays on the Home screen.
0 - The Skype for Business Search icon does not display on the Home screen.

Ribbon Communications Server
Ribbon Communications application server, also called EXPERiUS™ A2, provides full-featured, IP-based multimedia communications applications for business and consumers.

You can deploy EXPERiUS A2 as a standalone server or in combination with a Ribbon Communications CONTINUUM™ C20 server; features vary depending on your deployment.

The following features are available for phones registered with the Ribbon Communications servers:

• MADN-SCA—A shared group feature that provides support for conference barge in, privacy, and remote call appearance. MADN-SCA requires you to deploy EXPERiUS A2 and CONTINUUM C20 server.
• Global Address Book—The global address book (GAB) feature is a corporate directory application managed by the Ribbon Communications server.
• Personal Address Book—The personal address book (PAB) feature is managed by the Ribbon Communications server and allows multiple clients (phones, computer software) to read and modify a user’s personal directory of contacts. When one client changes a contact all other clients are immediately notified of the change by the Ribbon Communications server.
• E.911—Enhanced 911 services specific to Ribbon Communications C20 server implementation.

Multiple Appearance Directory Number - Single Call Appearance (MADN-SCA)
Multiple appearance directory number—single call appearance (MADN-SCA) enables a group of users to share a single directory number that displays as a single line to each member of the group.

When this feature is enabled, users can initiate or receive calls on this shared line. MADN-SCA requires you to deploy EXPERiUS A2 and CONTINUUM C20 server.
Only one call can be active on the line at a time on the MADN-SCA shared line. When a call is in progress, any incoming calls to the line receive a busy tone.

**MADN-SCA Parameters**

The following list includes all parameters available for configuring MADN-SCA and feature options.

**Note:** If you configure the line-specific parameter `reg.x.server.y.address`, you must also configure values in the line-specific parameter `reg.x.server.y.specialInterop`.

If you configure the global parameter `voIpProt.server.x.address`, you must also configure values in the global parameter `voIpProt.server.x.specialInterop`.

For all deployments, including Ribbon Communications, line-specific configuration parameters override global configuration parameters. If you set values in both line-specific and global parameters, line-specific parameters are applied and global parameters are not applied.

**reg.x.address**

The user part (for example, 1002) or the user and the host part (for example, `1002@polycom.com`) of the registration SIP URI.

Null (default)

string address

**reg.x.server.y.specialInterop**

Specify the server-specific feature set for the line registration.

Standard (Default)

GENBAND

ALU-CTS

ocs2007r2

lcs2005

**voIpProt.server.x.specialInterop**

Enables server-specific features for all registrations.

Standard (default)

All other phones = Standard, GENBAND, GENBAND-A2, ALU-CTS, DT, ocs2007r2, lcs2005

**reg.x.type**

Private (default) - Use standard call signaling.

Shared - Use augment call signaling with call state subscriptions and notifications and use access control for outgoing calls.

**reg.x.bargeInEnabled**

0 (default) - barge-in is disabled for line x.
1 - barge-in is enabled (remote users of shared call appearances can interrupt or barge in to active calls).

**reg.x.callsPerLineKey**

Set the maximum number of concurrent calls for a single registration x. This parameter applies to all line keys using registration x. If registration x is a shared line, an active call counts as a call appearance on all phones sharing that registration.

This per-registration parameter overrides `call.callsPerLineKey`.

24 (default)

1-24

**reg.x.auth.userId**

User ID to be used for authentication challenges for this registration.

Null (default)

string - If the User ID is non-Null, it overrides the user parameter entered into the Authentication submenu on the Settings menu of the phone.

**reg.x.auth.password**

The password to be used for authentication challenges for this registration.

Null (default)

string - It overrides the password entered into the Authentication submenu on the Settings menu of the phone.

**reg.x.outboundProxy.address**

The IP address or hostname of the SIP server to which the phone sends all requests.

Null (default)

IP address or hostname

**reg.x.auth.domain**

The domain of the authorization server that is used to check the user names and passwords.

Null (default)

string

**reg.x.thirdPartyName**

Null (default) - In all other cases.

string address - This field must match the `reg.x.address` value of the registration which makes up the part of a bridged line appearance (BLA).
Configuring Privacy on a MADN-SCA Line

In the UC Software download, Poly provides the following two sample enhanced feature key (EFK) macros that you can configure to display on the phone to change privacy states: `privacyReleaseRestoreESK.cfg` and `privacyEnableESK.cfg`.

When you set the line to shared, an incoming call alerts all the members of the group simultaneously, and the call can be answered by any group member. On the server, you can configure a privacy setting that determines whether or not, after the call is answered, other members of the group can barge in to the same call and whether or not a call on hold can be picked up by other members of the group.

Optionally, you can configure star codes on the server that you can dial on the phone to toggle the privacy setting during a single active call. Note the following call behavior. If the line is configured for privacy by default, you can use a star code to toggle privacy on and off during an active call. When the call ends, the line resets to privacy settings. If the line is configured on the server with privacy off, you can use a star code to toggle to privacy on during an active call but you cannot toggle back to privacy off during the call. When the call ends, the line resets to privacy off.

Example MADN-SCA Configuration

The following example configuration shows the minimum configuration you need to enable MADN-SCA on the phone.

Use the parameters illustrated in the example below.

Procedure

1. Enter values for the following parameters in a configuration file and save.
   
   The value `8630@polycom.com` is an example registration address.

2. Enter the name of the configuration file to the CONFIG_FILES field of the master configuration file and save.
Global Address Book (GAB)

Ribbon Communications global address book (GAB) is a read-only global directory set up by an administrator and can co-exist with other corporate directories on the phone.

Global Address Book Parameters

Use the parameters in the following list to configure this feature.

feature.corporateDirectory.alt.enabled

0 (default) - Disables the global address book service.
1 - Enables the global address book service.

dir.corp.alt.address

Enter the URL address of the GAB service provided by the server.
Null (default)
Hostname
FQDN

dir.corp.alt.port

Set the port that connects to the server if a full URL is not provided.
0 (default)
Null
1 to 65535

dir.corp.alt.user

Enter the user name used to authenticate to the Ribbon Communications server.
Null (default)
UTF-8 encoding string

dir.corp.alt.viewPersistence

Determine if the results from the last address directory search displays on the phone.
0 (default) - Disabled
1 - Enabled

dir.corp.alt.attribute.x.filter

Enter a filter to use to set a predefined search string through configuration files.
Null (default)
UTF-8 encoding string
**dir.corp.alt.attribute.x.sticky**

0 (default) – the filter string criteria for attribute x is reset after a reboot.

1 – the filter string criteria is retained through a reboot.

If you set an attribute to be sticky (set this parameter to 1), a "*" displays before the label of the attribute on the phone.

**dir.corp.alt.attribute.x.label**

Enter a label to identify a user.

Null (default)

UTF-8 encoding string

**dir.corp.alt.attribute.x.name**

Enter the name of the parameter to match on the server. Each name must be unique; however, a global address book entry can have multiple parameters with the same name. You can configure up to eight parameters (x = 1 to 8).

Null (default)

UTF-8 encoding string

**dir.corp.alt.attribute.x.type**

Define how x is interpreted by the phone. Entries can have multiple parameters of the same type.

first_name

last_name (default)

phone_number

SIP_address

Other – for display purposes only.

If the user saves the entry to the local contact directory on the phone, first_name, last_name, and phone_number are copied. The user can place a call to the phone_number and SIP_address from the global address book directory.

**dir.local.serverFeatureControl.method**

Specifies a method for synchronizing the directory and server.

None (default)

GENBANDSOPi - Enables the GENBANDSOPi protocol on the phone to get the personnel address book service from the Ribbon Communications server.

**Example GAB Configuration**

The following example shows the minimum parameters you need to configure to enable GAB on the phone.
Procedure

1. Enable GAB by configuring the values in `feature.corporateDirectory.alt` and `dir.corp.alt`.
   
   The following illustration includes an example GAB address book parameters in `dir.corp.alt.attribute`.

   ![Configuration Example]

   2. Save the configuration file.
   3. Enter the name of the configuration file to the CONFIG_FILES field of the master configuration file and save.

Personal Address Book (PAB)

The personal address book (PAB) enables users to read and modify a personal directory of contacts on their phone.

When users modify contact information using any soft client, desk phone, or mobile client registered to the same line, the change is made on all other clients, and users are notified immediately of the change by the Ribbon Communications server.

Personal Address Book Parameters

Use the parameters in the following list to configure this feature.

Note that when you enable server control, five telephone number fields per contact are available.

- `feature.corporateDirectory.alt.enabled`
  - 0 (default) - Disables the global address book service.
  - 1 - Enables the global address book service.

- `dir.local.serverFeatureControl.method`
  - Specifies a method for synchronizing the directory and server.
  - None (default)
  - GENBANDSOPI - Enables the GENBANDSOPI protocol on the phone to get the personnel address book service from the Ribbon Communications server.
dir.local.serverFeatureControl.reg

  Specifies the phone line to enable the personal address book feature on. 1 (default)
  1 -34

dir.genband.local.contacts.maxSize

  Specify the maximum number of contacts available in the Ribbon Communications personnel address book contact directory.

Example Personal Address Book Configuration

The following example shows an example PAB configuration.

Procedure

1. Enter the values shown for the following parameters and save the configuration file.

2. Enter the configuration file to the CONFIG_FILES field of the master configuration file and save.

Enhanced 911 (E.911) Location for Ribbon Communications

With the Enhanced 911 (E.911) feature, you can set the location of the phone for emergency calls on the phone or on the provisioning server. When the phone starts up, the phone prompts users to choose a location, which is stored on the phone. The location that users set for the phone is used to identify the phone location to 911 operators dispatching emergency services. This feature is available for all phones and is disabled by default only in a Ribbon Communications environment.

By default, users can make a 911 call when the phone is locked, regardless of the call state, or when other features are in use. When a 911 call is in progress, the call control option does not display, users cannot use the hard keys to control a call, and DND or call forwarding are disabled.

Enhanced 911 (E.911) Location Parameters for Ribbon Communications

Use the parameter below to configure this feature.

feature.E911.locationInfoSchema

  RFC4119 (default) - XML schema is used in Session Initiation Protocol (SIP) invite as per RFC4119 standard.
  RFC5139- XML schema is used in Session Initiation Protocol (SIP) invite as per RFC5139 standard.
**Manually Set the Phone’s Location**

Users can set their location for emergency calls on the phone.

**Procedure**

1. Register the phone.
2. The phone displays a warning message to set your location for 10 seconds.
3. Press the warning message to enter a location.
   If the warning message disappears, on the phone, go to **Settings > Status > Diagnostics > Warnings**.
4. Select **Details** to enter a location to the location tree navigation menu.
5. Choose a location and press **Save**.
6. On the phone, go to **Status > Location Information**.
   The location information displays in the Status menu.

**Emergency Instant Messages**

Configure audio alerts for incoming instant messages and set the duration of time that emergency messages display.

**Emergency Instant Message Parameters**

Use the following parameters to configure emergency messages on phones registered with Ribbon Communications.

- **feature.instantMessaging.displayTimeout**
  Specify the time in minutes instant messages display.
  Messages display until one of the following occurs:
  - Timeout
  - Another instant message is received
  - A pop-up message displays
  - The phone receives an incoming call
  - The user presses any key or message on the phone
  1 minute (default)
  1 – 60 minutes

- **feature.instantMessaging.ring**
  instantMessage (default) – The phone plays a configured tone when an emergency instant message is received.
  Silent – No tone is played.

- **feature.instantMessaging.enabled**
  0 (default) – The phone does not display emergency instant messages.
  1 - Received emergency instant messages display on the phone.
BroadSoft BroadWorks Server

This section shows you how to configure Poly devices with BroadSoft Server options.

Note that you cannot register lines with the BroadWorks R18 server and the R20 and later server on the same phone. All lines on the phone must be registered to the same BroadWorks server.

Some BroadSoft features require you to authenticate the phone with the BroadWorks XSP service interface as described in the section Authentication with BroadWorks Xtended Service Platform (XSP) Service Interface.

Authentication with BroadWorks Xtended Service Platform (XSP) Service Interface

You can configure Poly phones to use advanced features available on the BroadSoft BroadWorks server. The phones support the following advanced BroadSoft features:

- BroadSoft Enhanced Call Park
- Executive-Assistant
- BroadSoft UC-One directory, favorites, and presence
- BroadSoft UC-One personal call control features

To use these features on Poly devices with a BroadWorks server, you must authenticate the phone with the BroadSoft XSP service interface.

Authentication for BroadWorks XSP Parameters

The authentication method you use depends on which version of BroadWorks you are running.

If your server is running BroadWorks R19 or earlier, enable the following parameters to authenticate on the BroadWorks server using separate XSP credentials:

- `dir.broadsoft.xsp.address`
- `reg.x.broadsoft.userId`
- `reg.x.broadsoft.xsp.password`
- `reg.x.broadsoft.useXspCredentials`

If your server is running BroadWorks R19 Service Pack 1 or later, enable the following parameters to authenticate on the BroadWorks server using the same SIP credentials you used to register the phone lines:

- `dir.broadsoft.xsp.address`
- `reg.x.auth.userId`
- `reg.x.auth.password`
- `reg.x.broadsoft.userId`

See the following list for additional details on these parameters.

`reg.x.broadsoft.xsp.password`

Enter the password associated with the BroadSoft user account for the line. Required only when `reg.x.broadsoft.useXspCredentials=1`. 

Third-Party Servers
Null (default)

string

reg.x.broadsoft.userId
Enter the BroadSoft user ID to authenticate with the BroadSoft XSP service interface.
Null (default)

string

reg.x.broadsoft.useXspCredentials
If this parameter is disabled, the phones use standard SIP credentials to authenticate.
1 (default) - Use this value, if phone lines are registered with a server running BroadWorks R19 or earlier.
0 - Set to 0, if phone lines are registered with a server running BroadWorks R19 SP1 or later.

reg.x.auth.userId
User ID to be used for authentication challenges for this registration.
Null (default)

string - If the User ID is non-Null, it overrides the user parameter entered into the Authentication sub-menu on the Settings menu of the phone.

reg.x.auth.password
The password to be used for authentication challenges for this registration.
Null (default)

string - It overrides the password entered into the Authentication sub-menu on the Settings menu of the phone.

BroadWorks Call Decline Policy
For shared lines in a BroadSoft BroadWorks environment, you can enable users to reject calls to a shared line.

By default, users cannot reject calls to a shared line on Poly phones. When this feature is enabled and a user rejects a call to the shared line, the call is rejected on all phones registered with the shared line.

BroadWorks Call Decline Parameter
Use the parameter below to enable users to reject calls on a shared line.

call.shared.reject
For shared line calls on the BroadWorks server.
0 (default) - The Reject soft key does not display.
1 - The phone displays a Reject soft key to reject an incoming call to a shared line.
Flexible Seating

Flexible Seating enables a user of an assigned primary phone to simultaneously access a registered line as a guest from an alternate host phone.

The user's primary registration is active on the primary and host phone. Users can access the BroadSoft UC-One contact directory and favorites on the host phone, but the Poly contact directory and favorites are not available.

**Note:** Flexible Seating is different from the Hoteling feature in that it provides only the primary registration's label on the host phone without any synchronization of features or settings.

The following conditions apply to the Flexible Seating feature:

- The primary phone and host phone do not sync automatically, but you can manually sync the phones on the BroadSoft BroadWorks server.
- The phone configured for the host user cannot accept incoming calls. The host user can make only emergency outgoing calls that are defined by the BroadWorks server.
- If the Phone Lock feature is enabled, numbers defined in the authorized call list are not allowed for outgoing calls except the emergency numbers set on the BroadWorks server.
- The host user account is intended to be used as a placeholder account that supports guest users and is not intended to be assigned to an actual phone user.
- The guest user cannot change the user password when Flexible Seating is enabled for the phone.
  - You can change the host phone's user password from the Web Configuration Utility at any time.
  - You can change the host phone's user password from the phone screen only when the guest user is not logged in.

Flexible Seating is not compatible with the following features:

- Hoteling
- Visitor Desk Phone (VDP)
- User Profile Feature
- Local Call Forwarding
- Local DND

On the BroadWorks server, you can set a period of time when the server automatically logs out a user from a phone in case a user does not log out.

**Flexible Seating Parameters**

To configure a host phone to support the primary phone's line registration, you must configure a host user profile and a guest user profile on the BroadSoft BroadWorks server.

In the host user profile configuration files, add the configuration parameters shown in the following list and map these parameters to the corresponding BroadSoft BroadWorks configuration tags.

**hoteling.reg**

1 (default) - Specifies the phone line on the host phone which hosts the guest line.

**hotelingMode.type**

-1 (Default): The parameter does not exist on the BroadSoft server.
0 - Both Flexible Seating and Hoteling are disabled on the BroadSoft Device Management Server (DMS).
1 - Hoteling is enabled
2 - Flexible Seating is enabled but guest is not logged in.
3 - Flexible seating location is enabled and guest is logged in.

**Note:** This parameter overrides `voIpProt.SIP.specialEvent.checkSync.downloadDirectory` when set to 2 or 3.

### Guest Profile PIN
You can configure a PIN for each guest profile, which enables users to access their guest profile on a host phone using a PIN.

The PIN prevents other users from logging into a guest phone without the phone password or guest PIN. The guest profile PIN takes precedence over the local phone password and the guest user must log out of the phone with the PIN before another user can log in with their password.

### BroadSoft BroadWorks Configuration Tags
The following table shows the Poly parameters you can map to the corresponding BroadSoft tags.

<table>
<thead>
<tr>
<th>Poly Configuration Parameter</th>
<th>BroadSoft Tag</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>hoteling.reg</code></td>
<td><code>%BWHOTELINGLINE-x%</code></td>
</tr>
<tr>
<td><code>hotelingMode.type</code></td>
<td><code>%BWHOTELINGMODE-x%</code></td>
</tr>
</tbody>
</table>

### Executive-Assistant
Using configuration files, you can enable the BroadSoft Executive-Assistant feature on lines registered with the BroadWorks R20 or later server, and assign lines as an executive or an assistant.

Note that all corresponding Executive and Assistant lines must be registered to the same server.

After you enable the feature, users set as executives or assistants can set basic filters to control which calls are sent directly to an assistant to answer or sent to the executive first. Executives can also enable screening, which enables the executive's phone to display the incoming call notification for all filtered calls.

To use this feature on Poly phones registered with a BroadWorks server, you must authenticate the phone with the BroadSoft XSP service interface.

In addition, depending on the role you assign the user, the Executive or Assistant icon displays on the Home screen of the phone. You can also simplify the Executive and Assistant menus by adding or removing Pick Call and Barge-in soft keys from the menu.
Enhanced Feature Keys for Executive-Assistant Menus

You can create enhanced feature keys (EFK) to enable users to quickly access the Overview Executives menu for assistants or the Executive Settings menu for executives.

You can create an Executive or Assistant line key, soft key, or speed dial that displays on the Lines screen in addition to the feature icons that display by default on the Home screen.

When a user presses the Executive EFK on the executive’s phone, the Executive Settings menu displays, and when a user presses the Assistant EFK on the assistant's phone, the Overview Executives menu displays. You can configure a line or soft key for this feature using the following EFK macro:

- **Executive menu:** "$FExecutiveMenu$"
- **Assistant menu:** $FAssistantMenu$

Executive-Assistant Parameters

In the BroadWorks Web Portal, you must enable the Executive Service for private and shared executive lines, and the Executive-Assistant Service for private and shared assistant lines.

The BroadWorks server allows the following configuration options: Executive private line, Executive-Assistant Service line, and a shared alias line. Administrators can set up executive and assistant lines in the following scenarios:

- A private executive line with an assistant with a private line
- Shared executive line with an assistant with a private line
- Shared executive line with a shared line alias on the assistant’s phone
  - The shared line must be created as a shared location of a line with the Executive Service on the BroadWorks server.
  - In this option, the main line registration is a private line for the assistant, and the secondary registration is a shared line for the executive.

The following list includes the configuration parameters you can use to enable and configure the Executive-Assistant feature.

**feature.BSExecutiveAssistant.enabled**

- 0 (default) - Disables the BroadSoft Executive-Assistant feature.
- 1 - Enables the BroadSoft Executive-Assistant feature.

**feature.BSExecutiveAssistant.regIndex**

The registered line assigned to the executive or assistant for the BroadSoft Executive-Assistant feature.

- 1 (default) to 255 - The registered line for the Executive or Assistant.

**feature.BSExecutiveAssistant.userRole**

- ExecutiveRole (default) - Sets the registered line as an Executive line.
- AssistantRole - Sets the registered line as an Assistant line.

**Note:** A phone can only have a line set as an Executive or an Assistant; an Executive and an Assistant line can’t be on the same phone.
**feature.BSExecutiveAssistant.SimplifiedAssistant.enabled**

0 (default) - Displays the Pick Call and Barge-in soft keys in the Assistants menu on the phone.
1 - Removes the Pick Call and Barge-in soft keys from the Assistants menu on the phone.

**feature.BSExecutiveAssistant.SimplifiedExec.enabled**

0 (default) - Displays the Pick Call and Barge-in soft keys in the Assistants menu on the phone.
1 - Removes the Pick Call and Barge-in soft keys from the Assistants menu on the phone.

**Enhanced Call Park**

You can configure BroadWorks Enhanced Call Park per registered line.

The following features are available for Enhanced Call Park:

- You can configure Enhanced Call Park only using configuration files; you cannot configure the feature on the Web Configuration Utility or from the local phone interface.
- You can configure Enhanced Call Park for private lines and shared lines. No configuration is necessary to enable the call park notification for monitored BLF lines.
- The default star codes set for the `call.parkedCallRetrieveString` is `*88`.

**Enhanced Call Park Parameters**

The following list includes the configuration parameters you can use to enable and configure this feature.

**reg.x.enhancedCallPark.enabled**

0 (default) - To disable the BroadWorks Enhanced Call Park feature.
1 - To enable the BroadWorks Enhanced Call Park feature.

**reg.x.lineAddress**

The line extension for a shared line. This parameter applies to private lines and BroadSoft call park and retrieve. If there is no extension provided for this parameter, the call park notification is ignored for the shared line.

Null (default)

String

**feature.enhancedCallPark.allowAudioNotification**

0 (default) - Disables the audio notifications for parked calls on private and shared lines.
1 - Enables the audio notifications for parked calls on private and shared lines.

**call.parkedCallRetrieveString**

The star code that initiates retrieval of a parked call.

Null (default)

Permitted values are star codes.
BroadSoft Directory Support

The BroadSoft directories enable users to search and view their personal, group, or enterprise contacts. When the BroadSoft directories are integrated with Polycom BroadSoft UC-One Application, users can access the different types of directories and search for contacts. There are five types of BroadSoft Directories:

- **Enterprise Directory** – This directory enables users to search and view Active Directory global address list of an enterprise. Users can query by first name, last name, phone number, extension and mobile number, and access contact information.

- **Group Directory** – This directory enables users to view the contact details such as work, extension, and mobile numbers of contacts. Users can place a call to anyone in the user's group.

- **Group Common Directory** – This directory enables users to view the contact details such as names and phone numbers of common contacts listed in the Group Common Directory.

- **Enterprise Common Directory** – This directory enables users to view the contact details such as names and phone numbers of common contacts listed in the Enterprise Common Directory.

- **Personal Directory** – This directory enables users to view the contact details such as names and phone numbers of the contacts in the user's personal directory stored on the server. You must enable this feature to allow users to add, delete, or edit the contacts in the BroadSoft Personal Directory.

BroadSoft Directory Parameters

To perform a search and to view contacts on the BroadSoft directories, configure the directories. You can configure this feature using the parameters in the following list.

**feature.broadsoftGroupDir.enabled**

0 (default) - Disables Group Directory.

1 - Enables Group Directory.

**feature.broadsoftdir.enabled**

0 (default) - Disables Enterprise Directory.

1 - Enables Enterprise Directory.

Change causes system to restart or reboot.

**feature.broadsoftPersonalDir.enabled**

0 (default) - Disables Personal Directory.

1 - Enables Personal Directory.

Polycom BroadSoft UC-One Application

The Polycom BroadSoft UC-One application integrates with BroadSoft Enterprise Directory and BroadCloud services—a set of hosted services by BroadSoft—to provide the following features:

- **BroadSoft Directory** – Displays information for all users in the enterprise, for example, work and mobile phone numbers.
• **BroadSoft Self-Presence** – Displays the user’s aggregated presence received from the BroadSoft Messaging Server (UMS) on the phone.

• **BroadCloud Presence** – Enables users to share presence information with the BroadTouch Business Communicator (BTBC) client application.

• **BroadCloud Favorites** – Enables users to mark contacts as favorites with the BroadTouch Business Communicator (BTBC) client application.

Polycom’s BroadSoft UC-One application enables you to:
• Access the BroadSoft Directory
• Search for contacts in BroadSoft Directory
• View BroadSoft UC-One contacts and groups
• View the presence status of BroadSoft UC-One contacts
• View and filter BroadSoft UC-One contacts
• Activate and control BroadSoft UC-One personal call control features.

**BroadSoft UC-One Configuration Parameters**

The following list includes all parameters available to configure features in the BroadSoft UC-One application.

**feature.qml.enabled**

0 (default) - Disable the QML viewer on the phone. Note that the UC-One directory user interface uses QML as the user interface framework and the viewer is used to load the QML applications.

1 - Enable the QML viewer on phone.

Change causes system to restart or reboot.

**feature.broadsoftdir.enabled**

0 (default) - Disable simple search for Enterprise Directories.

1 - Enable simple search for Enterprise Directories.

Change causes system to restart or reboot.

**feature.broadsoftUcOne.enabled**

0 (default) - Disables the BroadSoft UC-One feature.

1 - Enables the BroadSoft UC-One feature.

Change causes system to restart or reboot.

**feature.presence.enabled**

0 (default) - Disable the presence feature – including buddy managements and user status.

1 - Enable the presence feature with the buddy and status options.

**homeScreen.UCOne.enable**

1 (default) - Enable the UC-One Settings icon to display on the phone Home screen.
0 - Disable the UC-One Settings icon to display on the phone Home screen.

**dir.broadsoft.xsp.address**
Set the IP address or hostname of the BroadSoft directory XSP home address.
Null (default)
IP address
Hostname
FQDN

**dir.broadsoft.xsp.username**
To set the BroadSoft Directory XSP home address.

**dir.broadsoft.xsp.password**
Set the password used to authenticate to the BroadSoft Directory XSP server.
Null (default)
UTF-8 encoding string

**xmpp.1.auth.password**
Specify the password used for XMPP registration.
Null (Default)
UTF-8 encoded string

**xmpp.1.dialMethod**
For SIP dialing, the destination XMPP URI is converted to a SIP URI, and the first available SIP line is used to place the call.
SIP (default)
String min 0, max 256

**xmpp.1.jid**
Enter the Jabber identity used to register with the presence server, for example:
presence.test2@polycom-alpha.eu.bc.im
Null (default)
String min 0, max 256

**xmpp.1.roster.invite.accept**
Choose how phone users receive the BroadSoft XMPP invitation to be added to a buddy list.
prompt (default) - phone displays a list of users who have requested to add you as a buddy and you can accept or reject the invitation.
Automatic
**xmpp.1.server**

Sets the BroadSoft XMPP presence server to an IP address, host name, or FQDN, for example: polycom-alpha.eu.bc.im.

Null (default)

dotted-decimal IP address, host name, or FQDN.

**xmpp.1.verifyCert**

Enable or disable verification of the TLS certificate provided by the BroadSoft XMPP presence server.

1 (default) - Enabled
0 - Disabled

**Configuring BroadSoft UC-One**

You can configure the UC-One Call Settings menu and feature options on the phone, in the Web Configuration Utility, and using configuration parameters.

**Configure BroadSoft UC-One on the Phone**

You can enable the BroadSoft UC-One feature directly from the phone.

**Procedure**

1. Navigate to **Settings > UC-One**.
2. Under General, click **Enable for BroadSoft UC-One**.
   
   This enables the UC-One Call Settings menu to display on the phone.

**Configure BroadSoft UC-One in the Web Configuration Utility**

You can enable the BroadSoft UC-One feature and feature options in the Web Configuration Utility.

**Procedure**

1. In the Web Configuration Utility, navigate to **Settings > UC-One**.
2. Under **Call Settings Features**, enable each feature menu you want available on the phone.

**BroadSoft UC-One Directory Parameters**

Use the parameters in the following list to configure the Polycom BroadSoft UC-One directory.

**dir.broadsoft.regMap**

Specify the registration line credentials you want to use for BroadSoft R20 Server or later to retrieve directory information from the BroadSoft UC-One directory when dir.broadsoft.useXspCredentials =0.

1 (default)

0 - Const_NumLineReg
dir.broadsoft.useXspCredentials

Specify which method of credentials the phone uses to sign in with the BroadSoft server.
1 (default) - Uses BroadSoft XSP credentials.
0 - Uses SIP credentials from dir.broadsoft.regMap.

Enterprise Directory Default Search
You can view an initial list of contacts in the Enterprise directory.
After you enable the feature, users can view a list of contacts by default without the need to enter a name in the search box of the directory.

Enterprise Directory Search Parameters
Use the following parameter to configure the Enterprise Directory Search feature.

feature.broadsoftdir.showDefaultSearch
0 (default) - No contacts are displayed when the search box field is empty.
1 - Enables the user to view the initial list of contacts for an empty search box

BroadSoft Server-Based Call Logs
You can configure the phone to view the list of call logs when the user taps the Recent soft key on the phone's screen.
When you enable this feature, users can view the call logs retrieved from the server on the phone.

BroadSoft Server-Based Call Logs Parameters
Use the following parameter to enable the BroadSoft server based call logs feature.

feature.broadsoft.callLogs
Disabled (default) - Disable the BroadSoft server call logs feature.
Basic - Enable the BroadSoft server call logs feature.

BroadSoft Server-Based Redial
You can configure the phone to support BroadSoft Server-Based Redial feature, which allows users to redial the last number dialed from any device connected to the same line or registration.
When enabled, the Redial soft key displays on the phone screen. Users can select this soft key to place a call to the last dialed number.

BroadSoft Server-Based Redial Parameter
Use the following parameter to configure this feature.

feature.broadsoft.basicCallLogs.redial.enabled
0 (default) - Disables the option to redial the last number.
Anonymous Call Rejection

Anonymous Call Rejection enables users to automatically reject incoming calls from anonymous parties who have restricted their caller identification.

After you enable the feature for users, users can turn call rejection on or off from the phone. When a user turns Anonymous Call Rejection on, the phone gives no indication that an anonymous call was received.

You can configure this option in the Web Configuration Utility.

Configure Anonymous Call Rejection using the Web Configuration Utility

You can configure Anonymous Call Rejection in the Web Configuration Utility.

Procedure

1. Navigate to Settings > UC-One.
2. Under the Call Setting Features, click Enable for Anonymous Call Rejection.

Anonymous Call Rejection Parameters

Use the parameters below to configure Anonymous Call Rejection Parameters.

Use the parameters in the following list to enable this feature.

feature.broadsoft.xsi.AnonymousCallReject.enabled

- 0 (default) - Does not display the Anonymous Call Rejection menu to users.
- 1 - Displays the Anonymous Call Rejection menu and the user can turn the feature on or off from the phone.

feature.broadsoftUcOne.enabled

- 0 (default) - Disables the BroadSoft UC-One feature.
- 1 - Enables the BroadSoft UC-One feature.

Change causes system to restart or reboot.

reg.x.broadsoft.userId

- Enter the BroadSoft user ID to authenticate with the BroadSoft XSP service interface.
- Null (default)
- string
Simultaneous Ring Personal

The Simultaneous Ring feature enables users to add phone numbers to a list of contacts whose phones ring simultaneously when the user receives an incoming call.

When you enable the display of the Simultaneous Ring menu option on the phone, users can turn the feature on or off from the phone and define which numbers should be included in the Simultaneous Ring group.

Simultaneous Ring Parameters

Use the parameters below to configure Simultaneous Ring.

Use the parameters in the following list to enable this feature.

feature.broadsoft.xsi.SimultaneousRing.enabled

- 0 (default) - Disables and does not display the Simultaneous Ring Personal feature menu on the phone.
- 1 - Enables the Simultaneous Ring Personal feature menu on the phone.

feature.broadsoftUcOne.enabled

- Enable or disable all BroadSoft UC-One features.
- 0 - Disabled
- 1 - Enabled

Line ID Blocking

You can enable or disable the display of the Line ID Blocking menu option on the phone.

When you enable the menu for users, users can choose to hide their phone number before making a call.

Line ID Blocking Parameters

Use the parameters below to configure Line ID Blocking.

Use the parameters in the following list to enable this feature.

feature.broadsoft.xsi.LineIDblock.enabled

- 0 (default) - Disables and does not display the Line ID Blocking feature menu on the phone.
- 1 - Enables the Line ID Blocking feature menu on the phone.

feature.broadsoftUcOne.enabled

- 0 (default) - Disables the BroadSoft UC-One feature.
- 1 - Enables the BroadSoft UC-One feature.

Change causes system to restart or reboot.
**BroadWorks Anywhere**

BroadWorks Anywhere enables users to use one phone number to receive calls to and dial out from their desk phone, mobile phone, or home office phone.

When you enable this feature, users can move calls between phones and perform phone functions from any phone. When enabled, the BroadWorks Anywhere settings menu displays on the phone and users can turn the feature on or off and add BroadWorks Anywhere locations on the phone.

**BroadWorks Anywhere Parameters**

You can configure BroadWorks Anywhere using configuration files or the Web Configuration Utility.

Use the parameters in the following list to enable this feature.

- `feature.broadsoft.xsi.BroadWorksAnywhere.enabled`
  - 0 (default) - Disables and does not display the BroadWorks Anywhere feature menu on the phone.
  - 1 - Enables the BroadWorks Anywhere feature menu on the phone.

- `feature.broadsoftUcOne.enabled`
  - 0 (default) - Disables the BroadSoft UC-One feature.
  - 1 - Enables the BroadSoft UC-One feature.
  - Change causes system to restart or reboot.

**BroadSoft Server-based Call Waiting**

You can configure the phone to support server-based call waiting, which enables the server to manage incoming calls while a user is in an active call.

When a user changes the call waiting state, the phone sends a request to the server to update to the new state. You can also configure the phone to specify the ringtone for incoming calls, when another call is in progress.

**BroadSoft Server-based Call Waiting Parameter**

Use the parameter below to configure server-based call waiting alerts.

- `feature.broadsoft.xsi.callWaiting.enabled`
  - 0 (default) - Disable incoming calls during an active call.
  - 1 - Enable incoming calls during an active call.

**Remote Office**

Remote Office enables users to set up a phone number on their office phone to forward incoming calls to a mobile device or home office number.

When enabled, this feature enables users to answer incoming calls to the office phone on the phone, and any calls placed from that phone show the office phone number.
Remote Office Parameters
Use the parameters in the following list to enable this feature.

feature.broadsoft.xsi.RemoteOffice.enabled
  0 (default) - Disables the Remote Office feature menu on the phone.
  1 - Enables and displays the Remote Office feature menu on the phone.

reg.x.broadsoft.userId
  Enter the BroadSoft user ID to authenticate with the BroadSoft XSP service interface.
  Null (default)
  string

feature.broadsoftUcOne.enabled
  0 (default) - Disables the BroadSoft UC-One feature.
  1 - Enables the BroadSoft UC-One feature.
  Change causes system to restart or reboot.

dir.broadsoft.xsp.password
  Set the password used to authenticate to the BroadSoft Directory XSP server.
  Null (default)
  UTF-8 encoding string

BroadSoft UC-One Credentials
Enabling this feature allows users to enter their BroadWorks UC-One credentials on the phone instead of in the configuration files.

The parameters reg.x.broadsoft.useXspCredentials, and feature.broadsoftUcOne.enabled must be enabled to display the UC-One Credentials menu option on the phone.

BroadSoft UC-One Credential Parameters
Use the parameters in the following list to enable this feature.

dir.broadsoft.xsp.address
  Set the IP address or hostname of the BroadSoft directory XSP home address.
  Null (default)
  IP address
  Hostname
  FQDN
**reg.x.broadsoft.userId**

Enter the BroadSoft user ID to authenticate with the BroadSoft XSP service interface.

Null (default)

string

**feature.broadsoftUcOne.enabled**

0 (default) - Disables the BroadSoft UC-One feature.

1 - Enables the BroadSoft UC-One feature.

Change causes system to restart or reboot.

**dir.broadsoft.xsp.username**

To set the BroadSoft Directory XSP home address.

**dir.broadsoft.xsp.password**

Set the password used to authenticate to the BroadSoft Directory XSP server.

Null (default)

UTF-8 encoding string

**feature.broadsoftdir.enabled**

0 (default) - Disable simple search for Enterprise Directories.

1 - Enable simple search for Enterprise Directories.

Change causes system to restart or reboot.

---

**BroadSoft Server-Based Call Forwarding**

To enable server-based call forwarding, you must enable the feature on both the server and the registered phone.

If you enable server-based call forwarding on one registration, other registrations are not affected.

The following conditions apply for server-based call forwarding:

- If server-based call forwarding is enabled, but inactive, when a user presses the **Forward** soft key, the moving arrow icon does not display on the phone and incoming calls are not forwarded.

The call server uses the Diversion field with a SIP header to inform the phone of a call's history. For example, when you enable call forwarding, the Diversion header allows the receiving phone to indicate who the call was from, and the phone number it was forwarded from.

---

**Hoteling**

The Hoteling feature enables users to log in to a guest profile to use any available shared phone.

After logging in, users have access to their own guest profile and settings on the shared phone. When hoteling is enabled, the Guest In soft key displays for users to log in to the phone.
**Note:** For additional details on configuring the hoteling feature, see *Using Hoteling on Polycom Phones: Feature Profile 76554* at Polycom Engineering Advisories and Technical Notifications.

---

### Hoteling Parameters

To enable Hoteling, you must configure Poly phones with the BroadSoft BroadWorks R17 platform.

You cannot use Hoteling in conjunction with the feature-synchronized automatic call distribution (ACD) feature and you must disable all ACD parameters to use the Hoteling feature. If both features are enabled at the same time, ACD take precedence and the Hoteling GuestIn/GuestOut soft keys do not display.

Use the parameters in the following list to configure Hoteling.

- **feature.hoteling.enabled**
  - 0 (default) - Enable Hoteling.
  - 1 - Disable Hoteling.

- **hoteling.reg**
  - Specify the line registration to use for Hoteling. You must disable the Automatic Call Distribution (ACD) feature and all ACD parameters to use Hoteling.
  - 1 (default)
  - 1 - 34

### Feature-Synchronized Automatic Call Distribution (ACD)

Feature-synchronized automatic call distribution (ACD) assists organizations in handling a large number of incoming phone calls to a call center with users in agent/supervisor roles.

Feature-synchronized ACD is distinct from and provides more advanced ACD functions than the Hoteling feature.

Feature-synchronized ACD is available in the following services.

- **Standard** – Standard service enables call center agents to sign in to a shared phone. When an agent is signed in, the phone displays the current state of the agent, for example, whether the agent is available or unavailable to take new calls.
- **Premium** – Premium service offers two additional features: Hoteling and Queue Status Notification.
  - Hoteling enables agents to use their agent credentials to log in to any available phone. If you want to enable the hoteling feature with feature-synchronized ACD, see the section Hoteling.
  - Queue status notification enables agents to view the queue status of a call center so that agents can adjust their call response.

The capabilities of this feature vary with the SIP call server. Consult your call server provider for information and for documentation. The SIP signaling used for this implementation is described in the BroadSoft BroadWorks document *Device Key Synchronization Requirements Document; Release R14 sp2; Document version 1.6.*
Note: For more information on standard and premium ACD as well as the hoteling and queue status notification enhancements, see Feature Profile 76179: Using Premium Automatic Call Distribution for Call Centers at Polycom Engineering Advisories and Technical Notifications.

ACD Agent Availability Parameters

Use the parameters in this list to configure ACD agent availability feature.

**feature.acdServiceControlUri.enabled**
Enable to display the Trace, Emergency, and Disp Code softkeys.
0 (default) - Disabled
1 - Enabled

**Note:** Set the parameter feature.enhancedFeatureKeys.enabled parameter to 1 to use the feature.acdServiceControlUri.enabled parameter.

**feature.acdLoginLogout.enabled**
0 (default) - Disables the ACD login/logout feature.
1 - Enables the ACD login/logout feature.

**feature.acdPremiumUnavailability.enabled**
0 (default) - Disables the premium ACD unavailability feature.
1 - Enables the premium ACD unavailability feature.

**voIPProt.SIP.acd.signalingMethod**
0 (default) - Support SIP-B signaling.
1 - Support synchronization signaling feature.

**acd.UnavailableMacroReasonCodeMenu.enabled**
0 (default) – Disables the display of unavailable reason code menu for unavailable macro.
1 - Enables the display of unavailable reason code menu for unavailable macro.

**Note:** If you set the acd.defaultUnavailReasonCode.enabled parameter to 0, then you must set acd.UnavailableMacroReasonCodeMenu.enabled parameter to 1.

**feature.showRejectSoftKey.enable**
1 (default) - Display Reject softkey for an incoming call.
0 - Doesn't display Reject softkey for an incoming call.
**reg.x.showRejectSoftKey**

1 (default) - Display Reject softkey for an incoming call on the configured registration line.
0 - Doesn't display Reject softkey for an incoming call on the configured registration line.

**Note:** If you configure `reg.x.ShowRejectSoftKey` parameter and `feature.showRejectSoftKey.enable` parameter, then the value configured for the `reg.x.ShowRejectSoftKey` parameter takes the precedence.

**acd.defaultUnavailReasonCode.enabled**

1 (default) - Display the reason code None in the unavailable reason code menu.
0 - Doesn't display the reason code None in the unavailable reason code menu.

**Note:** If you configure the `acd.defaultUnavailReasonCode.enabled` parameter to 0, then the first item in the Unavailable reason code list menu is a Non-select-able item.

**voIpProt.SIP.copyUnknownHeaders**

Specify the comma separated header names.

Default ()

String - The total number of headers is 15 and maximum number of characters is 256.

Example: `voIpProt.SIP.copyUnknownHeaders="User-to-User, x_TFN, PraestoSF-ID"`

---

### Configuring uaCSTA

When you configure Poly phones to use user agent Computer Supported Telecommunications Applications (uaCSTA) with a CSTA server, you can remotely control the phone and access phone services using a computer telephony integration (CTI) application on your computer.

Poly phones support two types of user agent configurations for CSTA:

- A dedicated line to control or monitor all the other lines on the phone.
- A single line to act as both SIP line and CSTA line.

Poly phones support the Minimum and Basic profiles compliant with “ECMA TR/087: Using CSTA for SIP Phone User Agents (uaCSTA).” For information, see [ECMA international](https://www.ecma-international.org/standards/).

**Note:** Poly phones do not support the Network Reached event.

Poly supports the following CSTA services and events:

#### CSTA Services

- MonitorStart
- MonitorStop
- MakeCall Without Prompt
• AnswerCall
• ClearConnection
• DeflectCall in alerting state
• HoldCall
• RetrieveCall
• SingleStepTransferCall
• SnapshotDevice
• Conference Call
• Transfer Call
• ConsultationCall

CSTA Events
• ServiceInitiated
• Originated
• Delivered
• Diverted
• Established
• ConnectionCleared
• Held
• Retrieved
• Failed
• Transferred
• BackInService
• OutOfService
• Conferenced

Capability Exchange Service
• GetSwitchingFunctionDevices

Capability Exchange Event
• SwitchingFunctionDevices

Enable uaCSTA as a Dedicated Line

You can configure one CSTA line on each phone. To ensure CSTA works correctly, Poly recommends that you configure the CSTA line as the last among all registered lines on the phone.

Procedure
1. Set up an account on your CSTA server.
2. Set the server to CSTA.
3. Enable the Poly per-registration parameter: \texttt{reg.x.csta=1}.

When you correctly register a CSTA line on a Poly phone, the CSTA line displays on the phone with an icon \includegraphics{csta_icon} and the default label \textbf{CSTA}. You can configure the label of the CSTA line.

If the CSTA line is not registered, an icon \includegraphics{unregistered_icon} shows that the line is unregistered.
A CSTA-registered line has no functionality to users. If a user selects a CSTA line on the phone, a message displays stating that no action is available.

**uaCSTA Parameters**

Use the following parameters to configure the uaCSTA feature.

You can use one CSTA line per phone.

**reg.x.csta**

Set the CSTA line x to the last registered line on the phone. A CSTA icon displays on the phone when this parameter is set to 1 and `reg.x.server.y.specialInterop = CSTA`.

0 (default) – Disable User Agent Computer Supported Telecommunications Applications (uaCSTA).

1 – Enable uaCSTA. This per-registration parameter overrides the global parameter `voIpProt.SIP.csta`.

**reg.x.server.y.specialInterop**

Specify the server-specific feature for the line registration.

When you set this parameter to CSTA and `reg.x.csta=1`, a CSTA icon displays on the phone. The line should receive the CSTA commands only.

Standard (Default)

If you configure `reg.x.csta="1"`, the phone works for single-line type CSTA.

**voIpProt.SIP.csta**

0 (default) – Disable uaCSTA.

1 – Enable uaCSTA. When enabled, `reg.x.csta overrides voIpProt.SIP.csta`.

**voIpProt.SIP.uaCSTA.deviceIDExt.enable**

0 – Disable (default)

1 – When you enable the parameter, the phone appends tel = "Reg_Label" to "deviceId" in CSTA SwitchingFunctionDevices service response.
Device Parameters

Topics:

- Changing Device Parameters
- Parameter List Conventions
- Device Parameters

The `<device/>` parameters (also known as device settings) contain default values you can use to configure settings for large-scale device deployments within your network.

Poly provides a global `device.set` parameter that you must enable to install software and change device parameters. Each `<device/>` parameter has a corresponding `.set` parameter that enables or disables the value for that device parameter. You need to enable the corresponding `.set` parameter for each parameter you want to apply.

After you complete the software installation or configuration changes using device parameters, remove `device.set` to prevent the phones from rebooting and triggering a reset of device parameters that phone users might have changed after the initial installation.

The `<device/>` parameters are designed to be stored in flash memory. Therefore, Poly phones don’t upload `<device/>` parameters to the `<MAC>-web.cfg` or `<MAC>-phone.cfg` override files that store settings you make using the system web interface (Web Configuration Utility) or the phone’s local interface. This design protects your ability to manage and access the phones using the standard set of parameters on a provisioning server after the initial software installation.

If you configure any parameter values using the `<device/>` parameters, subsequent configuration changes you make from the system web interface (Web Configuration Utility) or the phone’s local interface don’t take effect until after the phone reboots or restarts.

Changing Device Parameters

Note the following when modifying device parameters:

- You may ignore some parameters. For example, if you enable DHCP, that setting overrides the value you set with `device.net.ipAddress`.
- Although the phones check individual parameters to see whether they are in range, the phones don’t check the interaction between parameters. If a parameter is out of range, an error message displays in the log file, and the phones don’t use the parameter.
- Incorrect configuration can put the phones into a reboot loop. For example, server A has a configuration file that specifies that server B should be used, and server B has a configuration file that specifies that server A should be used.

To detect errors, including IP address conflicts, test new configuration files on two phones before deploying to all phones.
Types of Device Parameters

The following parameters outline the three types of `<device/>` parameters, their permitted values, and the default value.

**device.set**

0 (default) - Don’t use any `device.xxx` fields to set any parameters. Set this to 0 when you are not making changes to device parameters.

1 - Use the `device.xxx` fields that have `device.xxx.set=1`. Set this to 1 when you are making changes to device parameters.

Change may cause system to restart or reboot.

**device.xxx**

Configuration parameter.

String

Change may cause system to restart or reboot.

**device.xxx.set**

0 (default) - Don’t use the `device.xxx` value.

1 - Use the `device.xxx` value.

For example, if `device.net.ipAddress.set=1`, then use the value set for `device.net.ipAddress`.

Change may cause system to restart or reboot.

Parameter List Conventions

For each feature, Poly provides a list of parameters in XML that you can use to configure feature settings.

This guide documents parameters using parameter lists. Be sure to familiarize yourself with basic XML and parameter list conventions to successfully change configurations.

Using XML

Poly parameters are attributes of XML elements. Element names don't affect the behavior of parameters or operation of your phone, and you can customize as needed.

When configuring the parameters as XML, you must enter parameter names as attributes of a well-formed XML syntax. You can organize parameters into any well-formed XML element structure.

A `parameter="value"` pair is equivalent to an XML `attribute="value"` pair. For example:

```
<element1>
  <element2 feature.acousticFenceUI.enabled="1" />
</element1>
```
Understanding Parameter Lists

The following describes a general convention for details you can find in parameter lists. Parameter details can vary depending on the complexity of the parameter.

**parameter.name**

- A parameter’s description, applicability, or dependencies, as needed.
- The parameter’s permitted values, the default value, and the value’s unit of measure, such as seconds, Hz, or dB.
- An indication when a change in a parameter’s value causes a phone restart or reboot.

**Note:** A note that highlights critical information you need to know.

Parameter List Template and Examples

Parameter details can vary depending on the complexity of the parameter.

The following template shows the general parameter list conventions and details.

**parameter.name**

- A parameter’s description, applicability, or dependencies, as needed.
- The parameter’s permitted values, the default value, and the value’s unit of measure, such as seconds, Hz, or dB.
- An indication when a change in a parameter’s value causes a phone restart or reboot.

**Note:** A note that highlights critical information you need to know.

The following sample parameter lists show a few example parameters and some XML representations showing how to use them.

**feature.acousticFenceUI.enabled**

0 (default) - Hide the Acoustic Fence configuration setting on the phone.
1 - Display the Acoustic Fence configuration setting on the phone.
Change causes system to reboot or restart.

**XML Representation**

```xml
<element feature.acousticFenceUI.enabled="1" />
```

**reg.x.callsPerLineKey**

Set the maximum number of concurrent calls for a single registration x that you specify. This parameter applies to all line keys using registration x. If registration x is a shared line, an active call counts as a call appearance on all phones sharing that registration.

This per-registration parameter overrides the global parameter call.callsPerLineKey.

24 (default)
Device Parameters

Use the following `<device/>` parameters to configure some device settings.

**Note:** The default values for the `<device/>` parameters are set at the factory when the phones are shipped. For a list of the default values, see the latest Product Shipping Configuration Change Notice at Polycom Engineering Advisories and Technical Notifications.

device.auth.localAdminPassword

Set the phone’s local administrative password. The minimum length is defined by `sec.pwd.length.admin`.

string (32 character max)

device.auth.localUserPassword

Set the phone user’s local password. The minimum length is defined by `sec.pwd.length.user`.

string (32 character max)

device.auxPort.enable

Enable or disable the phone auxiliary port.

0 - Disable the phone auxiliary port

1 (default) - Enable the phone auxiliary port

Change causes system to restart or reboot.

device.baseProfile

NULL (default)

Generic - Sets the base profile to Generic for OpenSIP environments.

Lync - Sets this Base Profile for Skype for Business deployments.
**device.dhcp.bootSrvOpt**

When the boot server is set to Custom or Custom+Option66, specify the numeric DHCP option that the phone looks for.

- Null
- 128 to 254

Change causes system to restart or reboot.

**device.dhcp.bootSrvOptType**

Set the type of DHCP option the phone looks for to find its provisioning server if `device.dhcp.bootSrvUseOpt` is set to Custom.

- IP address - The IP address provided must specify the format of the provisioning server.
- String - The string provided must match one of the formats specified by `device.prov.serverName`.

Change causes system to restart or reboot.

**device.dhcp.bootSrvUseOpt**

- Default - The phone looks for option number 66 (string type) in the response received from the DHCP server. The DHCP server should send address information in option 66 that matches one of the formats described for `device.prov.serverName`.
- Custom - The phone looks for the option number specified by `device.dhcp.bootSrvOpt`, and the type specified by `device.dhcp.bootSrvOptType` in the response received from the DHCP server.
- Static - The phone uses the boot server configured through the provisioning server `device.prov.*` parameters.
- Custom and Default - The phone uses the custom option first or use Option 66 if the custom option is not present.

Change causes system to restart or reboot.

**device.dhcp.dhcpVlanDiscOpt**

Set the DHCP private option to use when `device.dhcp.dhcpVlanDiscUseOpt` is set to Custom.

- 128 to 254

Change causes system to restart or reboot.

**device.dhcp.dhcpVlanDiscUseOpt**

Set how VLAN Discovery occurs.

- Disabled - No VLAN discovery through DHCP.
- Fixed - Use predefined DHCP vendor-specific option values of 128, 144, 157 and 191 (`device.dhcp.dhcpVlanDiscOpt` is ignored).
- Custom - Use the number specified by `device.dhcp.dhcpVlanDiscOpt`.

Change causes system to restart or reboot.
**device.dhcp.enabled**
Enable or disable DHCP.

If the phone cannot communicate with the DHCP server, the phone's status bar will report "network down". The phone communicates with the DHCP server every five minutes to receive or validate the IP Address.

0 - DHCP is disabled.
1 (default) - DHCP is enabled.
Change causes system to restart or reboot.

**device.dhcp.option60Type**
Set the DHCP option 60 type.

Binary - Vendor-identifying information is in the format defined in RFC 3925.
ASCII - Vendor-identifying information is in ASCII format.
Change causes system to restart or reboot.

**device.dns.altSrvAddress**
Set the secondary server to which the phone directs domain name system (DNS) queries.

Server Address
Change causes system to restart or reboot.

**device.dns.domain**
Set the phone's DNS domain.

String
Change causes system to restart or reboot.

**device.dns.serverAddress**
Set the primary server to which the phone directs DNS queries.

Server Address
Change causes system to restart or reboot.

**device.hostname**
Specify a hostname for the phone when using DHCP by adding a hostname string to the phone's configuration.

If `device.host.hostname.set = 1, and device.host.hostname = Null, the DHCP client uses Option 12 to send a predefined host name to the DHCP registration server using Polycom_\<MACaddress>.

String — The maximum length of the host name string is ≤ 255 bytes, and the valid character set is defined in RFC 1035.
Change causes system to restart or reboot.
device.net.cdpEnabled
Determine if the phone attempts to determine its VLAN ID and negotiate power through CDP.
0 - Disabled
1 - Enabled
Change causes system to restart or reboot.

device.net.dot1x.anonid
EAP-TTLS and EAP-FAST only. Set the anonymous identity (user name) for 802.1X authentication.
String
Change causes system to restart or reboot.

device.net.dot1x.enabled
Enable or disable 802.1X authentication.
0 - Disabled
1 - Enabled
Change causes system to restart or reboot.

device.net.dot1x.identity
Set the identity (user name) for 802.1X authentication.
String
Change causes system to restart or reboot.

device.net.dot1x.method
Specify the 802.1X authentication method, where EAP-NONE means no authentication.
EAP-None
EAP-TLS
EAP-PEAPv0-MSCHAPv2
EAP-PEAPv0-GTC
EAP-TTLS-MSCHAPv2
EAP-TTLS-GTC
EAP-FAST
EAP-MD5

device.net.dot1x.password
Set the password for 802.1X authentication. This parameter is required for all methods except EAP-TLS.
String
Change causes system to restart or reboot.
**device.net.etherModeLAN**
Set the LAN port mode that sets the network speed over Ethernet.
Poly recommends that you don’t change this setting.
0 - Auto (default)
1 - 10HD
2 - 10FD
3 - 100HD
4 - 100FD
5 - 1000FD
HD means half-duplex and FD means full duplex.
Change causes system to restart or reboot.

**device.net.etherModePC**
Set the PC port mode that sets the network speed over Ethernet.
-1 - Disables the PC port
0 - Auto (default)
1 - 10HD
2 - 10FD
3 - 100HD
4 - 100FD
5 - 1000FD
HD means half-duplex and FD means full duplex.
Change causes system to restart or reboot.

**device.net.etherStormFilter**
1—DoS storm prevention is enabled and received Ethernet packets are filtered to prevent TCP/IP stack overflow caused by bad data or too much data.
0—DoS storm prevention is disabled.
Change causes system to restart or reboot.

**device.net.etherStormFilterPpsValue**
Set the corresponding packets per second (pps) for storm filter and to control the incoming network traffic.
17 to 40
38 (default)

**device.net.etherStormFilterPpsValue.set**
0 (default) - You can’t configure the device.net.etherStormFilterPpsValue parameter.
1 - You can configure the `device.net.etherStormFilterPpsValue` parameter.

**device.net.etherVlanFilter**

0 - Disabled
1 - Enabled

Change causes system to restart or reboot.

**device.net.ipAddress**

Set the phone’s IP address.

This parameter is disabled when `device.dhcp.enabled` is set to 1.

String

Change causes system to restart or reboot.

**device.net.IPgateway**

Set the phone’s default router.

IP address

Change causes system to restart or reboot.

**device.net.lldpEnabled**

0 - The phone doesn’t attempt to determine its VLAN ID.
1 - The phone attempts to determine its VLAN ID and negotiate power through LLDP.

Change causes system to restart or reboot.

**device.net.lldpFastStartCount**

Specify the number of consecutive LLDP packets the phone sends at the time of LLDP discovery, which are sent every one second.

5 (default)
3 to 10

**device.net.subnetMask**

Set the phone’s subnet mask.

This parameter is disabled when `device.dhcp.enabled` is set to 1.

subnet mask

Change causes system to restart or reboot.

**device.net.vlanId**

Set the phone’s 802.1Q VLAN identifier.

Null - No VLAN tagging.
0 to 4094
Device Parameters

Change causes system to restart or reboot.

**device.prov.maxRedunServers**
Set the maximum number of IP addresses to use from the DNS.
1 - 8
Change causes system to restart or reboot.

**device.prov.password**
Set the password for the phone to log in to the provisioning server, which may not be required.
If you modify this parameter, the phone re-provisions. The phone may also reboot if the configuration on the provisioning server has changed.
string
Change causes system to restart or reboot.

**device.prov.redunAttemptLimit**
Set the maximum number of attempts to attempt a file transfer before the transfer fails. When multiple IP addresses are provided by DNS, one attempt is considered to be a request sent to each server.
1 to 10
Change causes system to restart or reboot.

**device.prov.redunInterAttemptDelay**
Set the number of seconds to wait after a file transfer fails before retrying the transfer. When multiple IP addresses are returned by DNS, this delay only occurs after each IP has been tried.
0 to 300
Change causes system to restart or reboot.

**device.prov.serverName**
Enter the IP address, domain name, or URL of the provisioning server followed by an optional directory and optional configuration file name. This parameter is used if (device.dhcp.enabled is 0), if the DHCP server does not send a boot server option, or if the boot server option is static (device.dhcp.bootSrvUseOpt is static).
- IP address
- Domain name string
- URL
If you modify this parameter, the phone re-provisions. The phone also reboots if the configuration on the provisioning server has changed.

**device.prov.serverType**
Set the protocol the phone uses to connect to the provisioning server. Active FTP is not supported for BootROM version 3.0 or later, and only implicit FTPS is supported.
Device Parameters

FTP (default)
TFTP
HTTP
HTTPS
FTPS
Change causes system to restart or reboot.

device.prov.tagSerialNo

0 - The phone's serial number (MAC address) is not included in the User-Agent header of HTTPS/HTTPS transfers and communications to the microbrowser and web browser.
1 - The phone's serial number is included.

device.prov.upgradeServer

Specify the URL or path for a software version to download to the device.
On the system web interface, the path to the software version you specify displays in the drop-down list on the Software Upgrade page.
NULL (default)
string
0 - 255 characters

device.prov.user

The user name required for the phone to log in to the provisioning server (if required).
If you modify this parameter, the phone re-provisions, and it may reboot if the configuration on the provisioning server has changed.
string

device.sec.configEncryption.key

Set the configuration encryption key used to encrypt configuration files.
string
For more information, see the section Configuration File Encryption.
Change causes system to restart or reboot.

device.sec.coreDumpEncryption.enabled

Determine whether to encrypt the core dump or bypass the encryption of the core dump.
0 - Encryption of the core dump is bypassed.
1 (default) - the core dump is encrypted

device.sec.TLS.customCaCert1 ( TLS Platform Profile 1 )

Set the custom certificate to use for TLS Platform Profile 1 and TLS Platform Profile 2 and TLS Application Profile 1 and TLS Application Profile 2. The parameter
device.sec.TLS.profile.caCertList must be configured to use a custom certificate. Custom CA certificates can’t exceed 4096 bytes total size.

string
PEM format

device.sec.TLS.customDeviceCert1.privateKey
device.sec.TLS.customDeviceCert2.privateKey
Enter the corresponding signed private key in PEM format (X.509).
Size constraint: 4096 bytes for the private key.

device.sec.TLS.customDeviceCert1.publicCert
device.sec.TLS.customDeviceCert2.publicCert
Enter the signed custom device certificate in PEM format (X.509).
Size constraint: 8192 bytes for the device certificate.

device.sec.TLS.customDeviceCert1.set
device.sec.TLS.customDeviceCert2.set
Use to set the values for parameters
device.sec.TLS.customDeviceCertX.publicCert and
device.sec.TLS.customDeviceCertX.privateKey.
Size constraints are: 4096 bytes for the private key, 8192 bytes for the device certificate.

0 (default) - Disabled
1 - Enabled

device.sec.TLS.profile.caCertList1 ( TLS Platform Profile 1 )
Choose the CA certificate(s) to use for TLS Platform Profile 1 and TLS Platform Profile 2 authentication:
Builtin—The built-in default certificate
BuiltinAndPlatform—The built-in and Custom #1 certificates
BuiltinAndPlatform2—The built-in and Custom #2 certificates
All—Any certificate (built in, Custom #1 or Custom #2)
Platform1—Only the Custom #1 certificate
Platform2—Only the Custom #2 certificate
Platform1AndPlatform2—Either the Custom #1 or Custom #2 certificate

device.sec.TLS.profile.cipherSuite1 ( TLS Platform Profile 1 )
Enter the cipher suites to use for TLS Platform Profile 1 and TLS Platform Profile 2
string
device.sec.TLS.profile.cipherSuiteDefault1 (TLS Platform Profile 1)

Determine the cipher suite to use for TLS Platform Profile 1 and TLS Platform profile 2.
0 - The custom cipher suite is used.
1 - The default cipher suite is used.

device.sec.TLS.profile.deviceCert1 (TLS Platform Profile 1)

Choose the device certificate(s) for TLS Platform Profile 1 and TLS Platform Profile 2 to use for authentication.
Builtin
Platform1
Platform2

device.sec.TLS.profileSelection.dot1x

Choose the TLS Platform Profile to use for 802.1X.
PlatformProfile1
PlatformProfile2

device.sec.TLS.profileSelection.provisioning

Set the TLS Platform Profile to use for provisioning.
PlatformProfile1
PlatformProfile2
Change causes system to restart or reboot.

device.sec.TLS.profileSelection.syslog

Set the TLS Platform Profile to use for syslog.
PlatformProfile1
PlatformProfile2
Change causes system to restart or reboot.

device.sec.TLS.prov.strictCertCommonNameValidation

0 - Disables common name validation.
1 (default) - Provisioning server always verifies the server certificate for the commonName/
SubjectAltName match with the server hostname that the phone is trying to connect.

device.sec.TLS.syslog.strictCertCommonNameValidation

0
1 - Syslog always verifies the server certificate for the commonName/SubjectAltName match
with the server hostname that the phone is trying to connect.
device.sntp.gmtOffset
Set the GMT offset—in seconds—to use for daylight savings time, corresponding to -12 to +13 hours.
-43200 to 46800

device.sntp.gmtOffsetCityID
Sets the correct time zone location description that displays on the phone menu and in the Web Configuration Utility.
NULL (default)
0 to 126
For descriptions of all values, refer to Time Zone Location Description.

device.sntp.serverName
Enter the SNTP server from which the phone obtains the current time.
IP address
Domain name string

device.syslog.facility
Determine a description of what generated the log message.
0 to 23
For more information, see RFC 3164.

device.syslog.prependMac
0
1 - The phone’s MAC address is prepended to the log message sent to the syslog server.
Change causes system to restart or reboot.

device.syslog.renderLevel
Specify the logging level for the lowest severity of events to log in the syslog. When you choose a log level, the log includes all events of an equal or greater severity level, but it excludes events of a lower severity level.
0 or 1 - SeverityDebug(7).
2 or 3 - SeverityInformational(6).
4 - SeverityError(3).
5 - SeverityCritical(2).
6 - SeverityEmergency(0).
Change causes system to restart or reboot.

device.syslog.serverName
Set the syslog server IP address or domain name string.
IP address
Domain name string

device.syslog.transport
Set the transport protocol that the phone uses to write to the syslog server.
None - Transmission is turned off but the server address is preserved.
UDP
TCP
TLS
Quick Setup Soft Key Parameter

Use the following parameter to configure the Quick Setup soft key.

prov.quickSetup.enabled

0 (default) - Disables the quick setup feature.
1 - Enables the quick setup feature.
Per-Registration Call Parameters

Poly phones support an optional per-registration feature that enables automatic call placement when the phone is off-hook.

The phones also support a per-registration configuration that determines which events cause the missed-calls counter to increment. You can enable/disable missed call tracking on a per-line basis.

call.advancedMissedCalls.addToReceivedList
Applies to calls on that are answered remotely.
0 (default) - Calls answered from the remote phone are not added to the local receive call list.
1 - Calls answered from the remote phone are added to the local receive call list.

call.advancedMissedCalls.enabled
Use this parameter to improve call handling.
1 (default) - Shared lines can correctly count missed calls.
0 - Shared lines may not correctly count missed calls.

call.advancedMissedCalls.reasonCodes
Enter a comma-separated list of reason code indexes interpreted to mean that a call should not be considered as a missed call.
200 (default)

call.autoAnswer.micMute
1 (default) - The microphone is initially muted after a call is auto-answered.
0 - The microphone is active immediately after a call is auto-answered.

call.autoAnswer.ringClass
The ring class to use when a call is to be automatically answered using the auto-answer feature. If you set to a ring class with a type other than answer or ring-answer, the settings are overridden such that a ringtone of visual (no ringer) applies.
ringAutoAnswer (default)

call.autoAnswer.ringTone
Intercom (default) – Auto answer plays the intercom tone.
doubleBeep – Auto answer plays the double-beep tone.

call.autoAnswer.SIP
0 (default) - Disable auto-answer for SIP calls.
1 - Enable auto-answer for SIP calls.
call.autoAnswer.ringTone
   intercom (default) – While auto answering a call, phone plays an intercom tone.
   doubleBeep – Phone plays the double beep tone.

call.autoAnswerMenu.enable
   1 (default) - The Autoanswer menu displays and is available to the user.
   0 - The Autoanswer menu is disabled and is not available to the user.

call.BlindTransferSpecialInterop
   0 (default) - Do not wait for an acknowledgment from the transferee before ending the call.
   1 - Wait for an acknowledgment from the transferee before ending the call.

call.dialtoneTimeOut
   The time is seconds that a dial tone plays before a call is dropped.
   60 (default)
   0 - The call is not dropped.
   Change causes system to restart or reboot.

call.internationalDialing.enabled
   Use this parameter to enable or disable the key tap timer that converts a double tap of the
   asterisk (*) symbol to the plus (+) symbol used to indicate an international call.
   1 (default) - A quick double tap of * converts immediately to +. To enter a double asterisk (**),
   tap the asterisk (*) once and wait for the key tap timer to expire to enter a second asterisk (*).
   0 - You cannot dial plus (+) symbol and you must enter the international exit code of the country
   you are calling from to make international calls.
   This parameter applies to all numeric dial pads on the phone including for example, the contact
   directory.
   Change causes system to restart or reboot.

call.internationalPrefix.key
   0 (default)
   1

call.localConferenceEnabled
   1 (default) - The feature to join a conference during an active call is enabled and the Conference
   soft key displays.
   0 - The feature to join a conference during an active call is disabled and the Conference soft key
   doesn't display. When you try to join the Conference, an "Unavailable" message displays.
   Change causes system to restart or reboot.
**call.offeringTimeOut**

Specify a time in seconds that an incoming call rings before the call is dropped.

60 (default)

0 - No limit.

Note that the call diversion, no answer feature takes precedence over this feature when enabled.

Change causes system to restart or reboot.

**call.playLocalRingBackBeforeEarlyMediaArrival**

Determines whether the phone plays a local ring-back after receiving a first provisional response from the far end.

1 (default) - The phone plays a local ringback after receiving the first provisional response from the far end. If early media is received later, the phone stops the local ringback and plays the early media.

0 - No local ringback plays, and the phone plays only the early media received.

**call.playLocalRingBackBeforeEarlyMediaArrival**

0 (default) - URL mode is used for URL calls.

1 - Number mode is used for URL calls.

**call.ringBackTimeOut**

Specify a time in seconds to allow an outgoing call to remain in the ringback state before dropping the call.

60 (default)

0 - No limit.

Change causes system to restart or reboot.

**call.showDialpadOnProceeding**

0 (default) – The phone doesn't show the dialpad button while a placed call is outgoing.

1 – The phone displays the dialpad button while a placed call is outgoing.

**call.stickyAutoLineSeize**

0 (default) - Dialing through the call list uses the line index for the previous call. Dialing through the contact directory uses a random line index.

1 - The phone uses sticky line seize behavior. This helps with features that need a second call object to work with. The phone attempts to initiate a new outgoing call on the same SIP line that is currently in focus on the LCD. Dialing through the call list when there is no active call uses the line index for the previous call. Dialing through the call list when there is an active call uses the current active call line index. Dialing through the contact directory uses the current active call line index.

Change causes system to restart or reboot.
call.stickyAutoLineSeize.onHookDialing

0 (default)
If you set call.stickyAutoLineSeize to 1, this parameter has no effect. The regular stickyAutoLineSeize behavior is followed.
If you set call.stickyAutoLineSeize to 0 and set this parameter to 1, this overrides the stickyAutoLineSeize behavior for hot dial only. (Any new call scenario seizes the next available line.)
If you set call.stickyAutoLineSeize to 0 and set this parameter to 0, there is no difference between hot dial and new call scenarios.
A hot dial occurs on the line which is currently in the call appearance. Any new call scenario seizes the next available line.
Change causes system to restart or reboot.

call.switchToLocalRingbackWithoutRTP
Determine whether local ringback plays in the event that early media stops.
0 (default) – No ringback plays when early media stops.
1 – The local ringback plays if no early media is received.

call.teluri.showPrompt
1 (default) - Phone displays a pop-up box to either call or cancel the number when tel URI is executed.
0 - Phone doesn’t display the pop-up box.

Remote Packet Capture Parameters
Use these parameters to enable and set up the remote packet capture feature.

diags.dumpcore.enabled
Determine whether the phone generates a core file if it crashes.
1 (default) - The phone generates a core file.
0 - The phone doesn’t generate a core file.
Change causes system to restart or reboot.

diags.pcap.enabled
Enable or disable all on-board packet capture features.
0 (default) - Disable on-board packet capture features.
1 - Enable on-board packet capture features.

diags.pcap.remote.enabled
Enable or disable the remote packet capture server.
0 (default) - Disable the remote packet capture server.
1 - Enable the remote packet capture server.

diags.pcap.remote.password
Enter the remote packet capture password.
MAC Address>(default)
 alphanumeric value

diags.pcap.remote.port
Specify the TLS profile to use for each application.
2002 (default)
 Valid TCP Port

Per-Registration Dial Plan Parameters
All the following parameters are per-registration parameters that you can configure instead of the general equivalent dial plan parameters.
Per-registration parameters override the general parameters where x is the registration number; for example, dialplan.x.applyToTelUriDial overrides dialplan.applyToTelUriDial for registration x.

dialplan.userDial.timeOut
Specify the time in seconds that the phone waits before dialing a number entered while the phone is on hook.
Generic Base Profile (default) – 0
0-99 seconds
0-99 seconds
You can apply dialplan.userDial.timeOut only when its value is lower than up.IdleTimeOut .

dialplan.x.applyToCallListDial
Generic Base Profile (default) – 1
0 - The dial plan does not apply to numbers dialed from the received call list or missed call list, including sub-menus for this line.
1 - The dial plan applies to numbers dialed from the received call list or missed call list, including sub-menus for this line.
Change causes system to restart or reboot.

dialplan.x.applyToDirectoryDial
Generic Base Profile (default) – 1
0 - The dial plan is not applied to numbers dialed from the directory or speed dial, including auto-call contact numbers for this line.

1 - The dial plan is applied to numbers dialed from the directory or speed dial, including auto-call contact numbers for this line.

Change causes system to restart or reboot.

dialplan.x.applyToForward

Generic Base Profile (default) – 1

0 - The dial plan applies to forwarded calls for this line.

1 - The dial plan applies to forwarded calls for this line.

dialplan.x.applyToTelUriDial

0

1 (default)

Change causes system to restart or reboot.

dialplan.x.applyToUserDial

0

1 (default)

Change causes system to restart or reboot.

dialplan.x.applyToUserSend

0

1 (default)

Change causes system to restart or reboot.

dialplan.x.conflictMatchHandling

Selects the dialplan based on more than one match with the least timeout.

0 (default for Generic Profile)

1

dialplan.x.digitmap.timeOut

Generic Base Profile (default) – 0

Change causes system to restart or reboot.

dialplan.x.digitmap

Generic Base Profile (default) - Null

string - max number of characters 100

Change causes system to restart or reboot.
dialplan.x.e911dialmask
Null (default)
string - max number of characters 256

dialplan.x.e911dialstring
Null (default)
string - max number of characters 256

dialplan.x.impossibleMatchHandling
0 (default) - Digits are sent to the call server immediately.
1 - A reorder tone is played and the call is canceled.
2 - No digits are sent to the call server until the Send or Dial key is pressed.
3 - No digits are sent to the call server until the timeout is configured by
dialplan.X.impossibleMatchHandling.timeOut parameter.
Change causes system to restart or reboot.

dialplan.x.originaldigitmap
Null (default)
string - max number of characters 2560

dialplan.x.removeEndOfDial
0
1 (default)
Change causes system to restart or reboot.

dialplan.x.routing.emergency.y.server.z
0 (default)
1
2
3
x, y, and z = 1 to 3
Change causes system to restart or reboot.

dialplan.x.routing.emergency.y.value
Null (default)
string - max number of characters 64
Change causes system to restart or reboot.
**dialplan.x.routing.server.y.address**

Null (default)

string - max number of characters 256

Change causes system to restart or reboot.

**dialplan.x.routing.server.y.port**

5060 (default)

1 to 65535

Change causes system to restart or reboot.

**dialplan.x.routing.server.y.transport**

DNSnaptr (default)

TCPpreferred

UDPOnly

TLS

TCPOnly

Change causes system to restart or reboot.

---

**Local Contact Directory File Size Parameters**

Use the following parameters to set the size of the local contact directory.

The maximum local directory size is limited based on the amount of flash memory in the phone and varies by phone model. Configure a provisioning server that allows uploads to ensure a back-up copy of the directory when the phone reboots or loses power.

**dir.local.nonVolatile.maxSize**

Set the maximum file size of the local contact directory stored on the phone’s non-volatile memory.

1 - 100 KB

**dir.local.volatile**

0 (default) - The phone uses non-volatile memory for the local contact directory.

1 - Enables the use of volatile memory for the local contact directory.

**dir.local.volatile.maxSize**

Sets the maximum file size of the local contact directory stored on the phone’s volatile memory.

1 - 200 KB
## Parameter Elements for the Local Contact Directory

The following table describes each of the parameter elements and permitted values that you can use in the local contact directory.

### Local Contact Directory Parameter Elements

<table>
<thead>
<tr>
<th>Element</th>
<th>Definition</th>
<th>Permitted Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>fn</td>
<td>The contact's first name</td>
<td>UTF-8 encoded string of up to 40 bytes¹</td>
</tr>
<tr>
<td>ln</td>
<td>The contact's last name</td>
<td>UTF-8 encoded string of up to 40 bytes¹</td>
</tr>
<tr>
<td>ct</td>
<td>Contact Used by the phone to address a remote party in the same way that a user manually dials a string of digits or a SIP URL. Also used to associate incoming callers with a particular directory entry. The maximum field length is 128 characters.</td>
<td>UTF-8 encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL</td>
</tr>
<tr>
<td>sd</td>
<td>Speed Dial Index Associates a particular entry with a speed dial key for one-touch dialing or dialing.</td>
<td></td>
</tr>
<tr>
<td>lb</td>
<td>The label for the contact The label of a contact directory item is by default the label attribute of the item. If the label attribute does not exist or is Null, then the first and last names form the label. A space is added between first and last names.</td>
<td>UTF-8 encoded string of up to 40 bytes¹</td>
</tr>
</tbody>
</table>

¹ UTF-8 encoding is used for string values to ensure compatibility with different character sets and to avoid errors in transmission. UTF-8 is a variable-length character encoding that maps each character in a string to one to four bytes. It is commonly used on the internet and is backwards-compatible with many other encodings.
<table>
<thead>
<tr>
<th>Element</th>
<th>Definition</th>
<th>Permitted Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>pt</td>
<td>Protocol</td>
<td>SIP, H323, or Unspecified</td>
</tr>
<tr>
<td></td>
<td>The protocol to use when placing a call to this contact.</td>
<td></td>
</tr>
<tr>
<td>rt</td>
<td>Ring Tone</td>
<td>Null, 1 to 21</td>
</tr>
<tr>
<td></td>
<td>When incoming calls match a directory entry, this field specifies the ringtone to use.</td>
<td></td>
</tr>
<tr>
<td>dc</td>
<td>Divert Contact</td>
<td>UTF-8 encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL</td>
</tr>
<tr>
<td></td>
<td>The address to forward calls to if the Auto Divert feature is enabled.</td>
<td></td>
</tr>
<tr>
<td>ad</td>
<td>Auto Divert</td>
<td>0 or 1</td>
</tr>
<tr>
<td></td>
<td>If set to 1, callers that match the directory entry are diverted to the address specified for the divert contact element.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If auto-divert is enabled, it has precedence over auto-reject.</td>
<td></td>
</tr>
<tr>
<td>ar</td>
<td>Auto Reject</td>
<td>0 or 1</td>
</tr>
<tr>
<td></td>
<td>If set to 1, callers that match the directory entry specified for the auto reject element are rejected.</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> If auto divert is also enabled, it has precedence over auto reject.</td>
<td></td>
</tr>
<tr>
<td>Element</td>
<td>Definition</td>
<td>Permitted Values</td>
</tr>
<tr>
<td>---------</td>
<td>------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>bw</td>
<td>Buddy Watching</td>
<td>0 or 1</td>
</tr>
<tr>
<td></td>
<td>If set to 1, this contact is added to the list of watched phones.</td>
<td></td>
</tr>
<tr>
<td>bb</td>
<td>Buddy Block</td>
<td>0 or 1</td>
</tr>
<tr>
<td></td>
<td>If set to 1, this contact is blocked from watching this phone.</td>
<td></td>
</tr>
<tr>
<td>up</td>
<td>User Photo</td>
<td>1-24</td>
</tr>
</tbody>
</table>

### Feature Activation and Deactivation Parameters

Use the feature parameters to control the activation or deactivation of a feature at run time.

**feature.callCenterCallInformation.enable**

- 1 (default) - The phone displays a full-screen dialog showing call information details. The dialog closes after 40 seconds, or you can press **Exit** to close it and return to the active call screen. You can set how long the dialog displays using the parameter `up.idleTimeout`.
- 0 - The phone uses the active call screen, and ACD call information is not available.

**feature.callCenterStatus.enabled**

- 0 (default) - Disable the status event threshold capability.
- 1 - Enable the status event threshold capability to display at the top of the phone screen.

**feature.flexibleLineKey.enable**

- 0 (default) - Disables the Flexible Line Key feature.
- 1 - Enables the Flexible Line Key feature.

**feature.ringDownload.enabled**

- 1 (default) - The phone downloads ringtones when starting up.
- 0 - The phone does not download ringtones when starting up.

  Change causes system to restart or reboot.

**feature.uniqueCallLabeling.enabled**

- 0 (default) - Disable Unique Call Labeling.
- 1 - Enable Unique Call Labeling. Use `reg.x.line.y.label` to define unique labels.

  Change causes system to restart or reboot.
feature.urlDialing.enabled
1 (default) - URL/name dialing is available from private lines, and unknown callers are identified on the display by their phone’s IP address.
0 - URL/name dialing is not available.

reg.x.urlDialing.enabled
1 (default) - Enable dialing by URL for SIP registrations.
0 - Disable dialing by URL for SIP registrations.

HTTPD Web Server Parameters
The phone contains a local Web Configuration Utility server for user and administrator features.

Note: You can use several of these parameters with Microsoft Skype for Business Server. The parameter values have two default states: a generic default value and a different value when the phone is registered with Skype for Business Server. The default values are listed for both states where applicable.

The web server supports both basic and digest authentication. You can’t configure the authentication user name and password.

httpd.enabled
Base Profile = Generic
1 (default) - The web server is enabled.
0 - The web server is disabled.
Change causes system to restart or reboot.

httpd.cfg.enabled
Base Profile = Generic
1 (default) - The system web interface is enabled.
0 - The system web interface is disabled.
Change causes system to restart or reboot.

httpd.cfg.port
Port is 80 for HTTP servers. Take care when choosing an alternate port.
80 (default)
1 to 65535
Change causes system to restart or reboot.

httpd.cfg.secureTunnelPort
The port to use for communications when the secure tunnel is used.
443 (default)
1 to 65535
Change causes system to restart or reboot.

httpd.cfg.secureTunnelRequired
1 (default) - Access to the system web interface is allowed only over a secure tunnel (HTTPS) and non-secure (HTTP) is not allowed.
0 - Access to the system web interface is allowed over both a secure tunnel (HTTPS) and non-secure (HTTP).
Change causes system to restart or reboot.

Feature License Parameter
Use the following parameter to configure the feature licensing system.
Once you install a license on a phone, you can't remove it.

license.polling.time
Specifies the time (using the 24-hour clock) to check if the license has expired.
02:00 (default)
00:00 - 23:59
Change causes system to restart or reboot.

Chord Parameters
Chord sets are the sound effect building blocks that use synthesized audio instead of sampled audio.
Poly phones support three chord sets. Each chord set has different chord names, represented by $x$ in the following parameters.

- callProg, where $x$ can be one of the following chord names:
  - dialTone
  - busyTone
  - ringback
  - reorder
  - stutter_3
  - callWaiting
  - callWaitingLong
  - howler
  - recWarning
  - stutterLong
• intercom
• callWaitingLong
• precedenceCallWaiting
• preemption
• precedenceRingback
• spare1 to spare6

• misc, where x can be one of the following chord names:
  • spare1 to spare9
  • cs1 to cs12

• ringer, where x can be one of the following chord names:
  • ringback
  • originalLow
  • originalHigh
  • spare1 to spare19

\[\text{tone.chord.callProg.x.freq.y} \quad \text{tone.chord.misc.x.freq.y} \quad \text{tone.chord.ringer.x.freq.y}\]

Frequency (in Hertz) for component y. Up to six chord-set components can be specified (y=1 to 6).

\[0-1600\]

\[\text{tone.chord.callProg.x.level.y} \quad \text{tone.chord.misc.x.level.y} \quad \text{tone.chord.ringer.x.level.y}\]

Level of component y in dBm0. Up to six chord-set components can be specified (y=1 to 6).

\[-57 \text{ to } 3\]

\[\text{tone.chord.callProg.x.onDur} \quad \text{tone.chord.misc.x.onDur} \quad \text{tone.chord.ringer.x.onDur}\]

On duration (length of time to play each component) in milliseconds.

\[0=\text{infinite} \quad \text{Positive integer}\]

\[\text{tone.chord.callProg.x.offDur} \quad \text{tone.chord.misc.x.offDur} \quad \text{tone.chord.ringer.x.offDur}\]

Off duration (the length of silence between each chord component) in milliseconds

\[0=\text{infinite} \quad \text{Positive integer}\]
tone.chord.callProg.x.repeat tone.chord.misc.x.repeat
tone.chord.ringer.x.repeat

Number of times each ON/OFF cadence is repeated.

0=infinte
Positive integer

**Message Waiting Parameters**

Use the following parameters to configure the message-waiting feature, supported on a per-registration basis.

The maximum number of registrations (x) for each phone model is listed in the Flexible Call Appearances section under the column "Registrations."

**msg.bypassInstantMessage**

0 (default) - Displays the Message Center and Instant Messages menus when a user presses the Messages or MSG key.

1 - Bypasses the menus and goes to voicemail.

**msg.mwi.x.led**

1 (default) - The LED flashes as long as the phone has new unread voicemail messages for any line.

0 - Red MWI LED doesn't flash when there are new unread messages for the selected line.

x is an integer referring to the registration indexed by reg.x.

**mwi.sharedLineIcon.enable**

1 (default) – Shows that the message waiting indicator appears for all the registered lines.

0 – The message waiting indicator shows only for the first line appearance if there are multiple lines registered on the phone.

**Ethernet Interface MTU Parameters**

Use the following parameters to control the Ethernet interface maximum transmission unit (MTU).

**net.interface.mtu**

Configures the Ethernet or Wi-Fi interface maximum transmission unit (MTU).

1496 (default)
800 - 1500

This parameter affects the LAN port and the PC port.
**net.interface.mtu6**

Specifies the MTU range for IPv6.

- 1500 (default)
- 1280 - 1500

**net.lldp.extenedDiscovery**

Specifies the duration of time that LLDP discovery continues after sending the number of packets defined by the parameter `lldpFastStartCount`.

- 0 (default)
- 0 - 3600

The LLDP packets are sent every 5 seconds during this extended discovery period.

**Presence Parameters**

Use the following parameters to configure the presence feature.

Note that the parameter `pres.reg` is the line number used to send SUBSCRIBE. If this parameter is missing, the phone uses the primary line to send SUBSCRIBE.

**pres.idleTimeoutoffHours.enabled**

- 1 (default) - Enables the off hours idle timeout feature.
- 0 - Disables the off hours idle timeout feature.

**pres.idleTimeoutoffHours.period**

The number of minutes to wait while the phone is idle during off hours before showing the Away presence status.

- 15 (default)
- 1 - 600

**pres.idleTimeout.officeHours.enabled**

- 1 (default) - Enables the office hours idle timeout feature
- 0 - Disables the office hours idle timeout feature

**pres.idleTimeout.officeHours.periods**

The number of minutes to wait while the phone is idle during office hours before showing the Away presence status.

- 15 (default)
- 1 - 600
Provisioning Parameters

Use the following parameters to control the provisioning server system for your phones.

**prov.autoConfigUpload.enabled**

1 (default) - Enables the automatic upload of configuration files from the phone or Web configuration utility to the provisioning server.

0 - Disabled the automatic upload of configuration files from the phone or Web configuration utility to the provisioning server.

**prov.configUploadPath**

Specifies the directory path where the phone uploads the current configuration file.

Null (default)

String

**prov.login.lcCache.domain**

The user's domain name to sign in.

Null (default)

String

**prov.login.lcCache.user**

The user's sign-in name to log in.

Null (default)

String

**prov.login.password.encodingMode**

The default encoding mode for the text in the Password field on the User Login screen.

123 (default)

Alphanumeric

**prov.login.userId.encodingMode**

The default encoding mode for the text in the User ID field on User Login screen.

Abc (default)

Alphanumeric

**prov.loginCredPwdFlushed.enabled**

1 (default) - Resets the password field when the user logs in or logs out.

0 - Does not reset the password field when the user logs in or logs out.
prov.startupCheck.enabled

1 (default) - The phone is provisioned on startup.
0 - The phone is not provisioned on startup.

prov.quickSetup.limitServerDetails

0 (default) - Provide all the necessary details for the given fields.
1 - Enter only the user name and password fields. Other details are taken from ztp/dhcp (option66).

Configuration Request Parameter

Use the following parameter to configure the phone's behavior when it receives a request for restart or reconfiguration.

request.delay.type

Specifies whether the phone should restart or reconfigure.

call (default) - The phone executes the request when there are no calls.
audio - The phone executes the request when there is no active audio.
Change causes system to restart or reboot.

User Preferences Parameters

Use the following parameters to set phone user preferences.

up.backlight.idleIntensity

Brightness of the LCD backlight when the phone is idle. Range is 0 to 3.

1 (Default) - Low
0
2 - Medium
3 - High

If this setting is higher than active backlight brightness (onIntensity), the active backlight brightness is used.

up.backlight.onIntensity

Brightness of the LCD backlight when the phone is active (in use). Range is 0 to 3.

3 (Default) - High
1 - Low
2 - Medium
up.backlight.timeout
Number of seconds to wait before the backlight dims from the active intensity to the idle intensity. Range is 5 to 60.
40 (default)

up.basicSettings.networkConfigEnabled
Specifies whether Network Configuration is shown or not shown under the Basic Settings menu.
0 (default) - Network Configuration is not shown under Basic Settings.
1 - Basic Settings menu shows Network Configuration with configurable network options for the user without administrator rights.

up.DIDFormat
NumberAndExtension (default) – Display the DID number and extension.
NumberOnly – Display the DID number on the phone screen.

up.cfgWarningsEnabled
Specifies whether a warning displays on a phone or not.
0 (Default) - Warning does not display.
1 - Warning is displayed on the phone if it is configured with pre-UC Software 3.3.0 parameters.

up.formatPhoneNumbers
Enable or disable automatic number formatting.
1 (Default)
0

up.hearingAidCompatibility.enabled
Specifies whether audio Rx equalization is enabled or disabled.
0 (Default) - Audio Rx equalization is enabled.
1 - Phone audio Rx (receive) equalization is disabled for hearing aid compatibility.

up.idleRestingState
menu (default) – The idle screen will display the Home screen menu.
calendar – The idle screen will display a top-level calendar.
dialpad – The idle screen will display a dial pad

up.idleStateView
Sets the phone default view.
0 (Default) - Call/line view is the default view.
1 - Home screen is the default view.
Change causes system to restart or reboot.

**up.idleTimeout**
Set the number of seconds that the phone is idle for before automatically leaving a menu and showing the idle display.
During a call, the phone returns to the Call screen after the idle timeout.
40 seconds (default)
0 to 65535 seconds
Change causes system to restart or reboot.

**up.IdleViewPreferenceRemoteCalls**
Determines when the phone displays the idle browser.
0 (Default) - Phone with only remote calls active, such as on a BLF monitored line, is treated as in the idle state and the idle browser displays.
1 - Phone with only remote calls active, such as on a BLF monitored line, is treated as in the active state and the idle browser does not display.
Change causes system to restart or reboot.

**up.lineKeyCallTerminate**
Specifies whether or not you can press the line key to end an active call.
0 (Default) - User cannot end an active call by pressing the line key.
1 - User can press a line key to end an active call.

**up.numberFirstCID**
Specifies what is displayed first on the **Caller ID** display.
0 (Default) - **Caller ID** display shows the caller’s name first.
1 - Caller’s phone number is shown first.
Change causes system to restart or reboot.

**up.numOfDisplayColumns**
Sets the maximum number of columns on the display. Set the maximum number of columns that phones display. Range is 0 to 4.
3 (Default)
0 - Phones display one column.
Change causes system to restart or reboot.

**up.osdIncomingCall.Enabled**
Specifies whether or not to display full screen popup or OSD for incoming calls.
1 (Default) - Full screen popup or OSD for incoming calls displays.
0 - Full screen popup or OSD for incoming calls does not display.
**up.rebootSoundEnabled**

1 (default) – Enable a sound effect alert when the phone reboots.

0 – Disable a sound effect alert when the phone reboots.

**up.ringer.minimumVolume**

Configure the minimum ringer volume. This parameter defines how many volume steps are accessible below the maximum level by the user.

16 (Default) - Full 16 steps of volume range are accessible.

0 - Ring volume is not adjustable by the user and the phone uses maximum ring volume.

Example: Upon bootup, the volume is set to ½ the number of configured steps below the maximum (16). If the parameter is set to 8 on bootup, the ringer volume is set to 4 steps below maximum.

**up.screenSaver.enabled**

0 (Default) - Screen saver feature is disabled.

1 - Screen saver feature is enabled. If a USB flash drive containing images is connected to the phone, and the idle browser is not configured, a slide show cycles through the images from the USB flash drive when the screen saver feature is enabled.

The images must be stored in the directory on the flash drive specified by **up.pictureFrame.folder**. The screen saver displays when the phone has been in the idle state for the amount of time specified by **up.screenSaver.waitTime**.

**up.screenSaver.type**

Choose the type of screen saver to display.

0 (Default) - Phone screen saver displays default images.

2 - Phone screen saver displays the idle browser.

**up.screenSaver.waitTime**

Number of minutes that the phone waits in the idle state before the screen saver starts. Range is 1 to 9999 minutes.

15 (Default)

**up.simplifiedSipCallInfo**

1 (Default) - This displayed host name is trimmed for both incoming and outgoing calls and the protocol tag/information is not displayed for incoming and outgoing calls.

0 - The full host name displays and the protocol tag/information displays for incoming and outgoing calls.

**up.softkey.transferTypeOption.enabled**

1 - The user can change the transfer type from consultative to blind and vice versa using a soft key after the user has initiated a transfer, but before completing the call to the far end.
0 (default) - There is no option to change from consultative to blind and blind to consultative when the user is in dial prompt after pressing the Transfer soft key.

**up.status.message.flash.rate**
Controls the scroll rate of the status bar. Range is 2 to 8 seconds.
2 seconds (Default)

**up.showDID**
AllScreens (default) – Display the DID number on all the screens.
None – Disable DID number on phone.
LockedScreen – Display the DID number on the lock screen.
StatusScreen – Display the DID number on the Status screen/Idle screen.
IncomingOSD – Display the DID number on the incoming On Screen Display (OSD) screen.
LockedScreenIncomingOSD – Display the DID number on the lock and incoming OSD screen.
LockedAndStatusScreen – Display the DID number on the lock and Status/Idle screen.
StatusScreenIncomingOSD – Display the DID number on the incoming OSD and Status/Idle screen.

**up.volumeChangeTone.enabled**
1 (default) – The phone plays a tone when the user adjusts the ringer or call volume.
0 – The phone does not play a tone.
Zoom Rooms Base Profile: 0 (default)

**up.warningLevel**
Line keys block display of the background image. All warnings are listed in the Warnings menu.
0 (Default) - The phone's warning icon and a pop-up message display on the phone for all warnings.
1 - Warning icon and pop-up messages are only shown for critical warnings.
2 - Phone displays a warning icon and no warning messages. For all the values, all warnings are listed in the Warnings menu.
Access to the Warnings menu varies by phone model.
Change causes system to restart or reboot.

**up.welcomeSoundEnabled**
1 (Default) - Welcome sound is enabled and played each time the phone reboots.
0 - Welcome sound is disabled.
To use a welcome sound you must enable the parameter up.welcomeSoundEnabled and specify a file in saf.x. The default UC Software welcome sound file is Welcome.wav.
Change causes system to restart or reboot.
**up.welcomeSoundOnWarmBootEnabled**

0 (Default) - Welcome sound is played when the phone powers on (cold boot), but not after it restarts or reboots (warm boot).

1 - Welcome sound plays each time the phone powers on, reboots, or restarts.

Change causes system to restart or reboot.

**up.display.showFullCallerID**

Phone displays the caller ID.

0 (default) – Phone displays the caller ID on the first line.

1 – Phone displays the caller ID on the second line.

**up.answerCall.listOrder**

Defines the order to answer a call upon pressing speaker button on the phone.

LIFO (default) - Last-In, First-Out.

FIFO - First-In, First-Out.

**Upgrade Parameters**

Specify the URL of a custom download server and the UC Software download server when you want the phone to check when to search for software upgrades.

**upgrade.custom.server.url**

The URL of a custom download server.

URL (default) - NULL

**upgrade.plcm.server.url**

The URL of the UC Software software download.

URL - http://downloads.polycom.com/voice/software/

**Voice Parameters**

Use the following parameters to configure phone audio.

**voice.rxPacketFilter**

Define a high-pass filter to improve sound intelligibility when the phone receives narrow band signals. Narrow band signals occur when a narrow band codec is in use, such as G.711mu, G.711A, G.729AB, iLBC, and some Opus and SILK variants.

0 (default) - Pass through.

1 - 300 Hz high-pass.
2 - 300 Hz high-pass with pre-emphasis. Use this value with G.729.

**voice.txPacketDelay**

Null (default)

normal, Null - Audio parameters are not changed.

low - If there are no precedence conflicts, the following changes are made:

```plaintext
voice.codecPref.G722="1"
voice.codecPref.G711Mu="2"
voice.codecPref.G711A="3"
voice.codecPref.<OtherCodecs>="
voice.audioProfile.G722.payloadSize="10"
voice.audioProfile.G711Mu.payloadSize= "10"
voice.audioProfile.G711A.payloadSize= "10"
voice.aec.hs.enable="0"
voice.ns.hs.enable="0"
```

Change causes system to restart or reboot.

**voice.txPacketFilter**

Null (default)

0 - Tx filtering is not performed.

1 - Enables Narrowband Tx high pass filter.

Change causes system to restart or reboot.

**Acoustic Echo Suppression (AES) Parameter**

Use the following parameter to enable speakerphone acoustic echo suppression (AES).

This feature removes residual echo after AEC processing. Because AES depends on AEC, enable AES only when you also enable AEC using `voice.aec.hd.enable`.

**voice.aes.hs.enable**

1 (default) - Enables the handset AES function.

0 - Disables the handset AES function.

**Comfort Noise Parameters**

Use the following parameters to configure the addition and volume of comfort noise during conferences.

**voice.cn.hf.enable**

0 (default) - Comfort noise not added.

1 - Adds comfort noise added into the Tx path for hands-free operation.

Far end users should use this feature when they find the phone to be 'dead', as the near end user stops talking.
**voice.cn.hf.attn**

35 (default) - quite loud
0 - 90

Attenuation of the inserted comfort noise from full scale in decibels; smaller values insert louder noise. Use this parameter only when `voice.cn.hf.enabled` is 1.

**voice.cn.hd.attn**

30 (default) - quite loud
0 - 90

Attenuation of the inserted comfort noise from full scale in decibels; smaller values insert louder noise. Use this parameter only when `voice.cn.hd.enabled` is 1.

**voice.cn.hs.enable**

0 (default) - Comfort noise is not added into the Tx path for the handset.
1 - Adds comfort noise is added into the Tx path for the headset.

Far end users should use this feature when they find the phone to be 'dead', as the near end user stops talking.

**voice.cn.hs.attn**

35 (default) - quite loud
0 - 90

Attenuation of the inserted comfort noise from full scale in decibels; smaller values insert louder noise. Use this parameter only when `voice.cn.hs.enabled` is 1.

**voice.vadRxGain**

Tunes VAD or CNG interoperability in a multi-vendor environment.

0 (default)
-20 to +20 dB

The specified gain value in dB is added to the noise level of an incoming VAD or CNG packet, when in a narrow band call.

When tuning in multi-vendor environments, the existing Poly to Poly phone behavior can be retained by setting `voice.vadTxGain = -voice.vadRxGain`.

This parameter is ignored for HD calls.

**voice.vadTxGain**

Tunes VAD or CNG interoperability in a multi-vendor environment.

0 (default)
-20 to +20 dB

The specified gain value in dB is added to the noise level of an incoming VAD or CNG packet, when in a narrow band call.
This causes the noise level to synthesize at the local phone to change by the specified amount.
When tuning in multi-vendor environments, the existing Poly to Poly phone behavior can be retained by setting `voice.vadTxGain = -voice.vadRxGain`.
This parameter is ignored for HD calls.

**Voice Jitter Buffer Parameters**

Use the following parameters to configure wired network interface voice traffic and push-to-talk interface voice traffic.

**voice.rxQoS avgJitter**

The average jitter in milliseconds for wired network interface voice traffic.

- **20 (default)**
- **0 to 80**

`avgJitter`: The wired interface minimum depth will be automatically configured to adaptively handle this level of continuous jitter without packet loss.
Change causes system to restart or reboot.

**voice.rxQoS maxJitter**

The average jitter in milliseconds for wired network interface voice traffic.

- **240 (default)**
- **0 to 320**

`maxJitter`: The wired interface jitter buffer maximum depth will be automatically configured to handle this level of intermittent jitter without packet loss.
Actual jitter above the average but below the maximum may result in delayed audio play out while the jitter buffer adapts, but no packets are lost. Actual jitter above the maximum value always results in packet loss. If legacy `voice.audioProfile.x.jitterBuffer.*` parameters are explicitly specified, they are used to configure the jitter buffer and these `voice.rxQoS` parameters are ignored.
Change causes system to restart or reboot.

**voice.rxQoS.prt avgJitter**

The average jitter in milliseconds for IP multicast voice traffic.

- **150 (default)**
- **0 - 200**

`avgJitter`: The PTT/Paging interface minimum depth is automatically configured to adaptively handle this level of continuous jitter without packet loss.
Change causes system to restart or reboot.

**voice.rxQoS.prt maxJitter**

The maximum jitter in milliseconds for IP multicast voice traffic.

- **480 (default)**
20 - 500

**maxJitter**: The PTT/Paging interface jitter buffer maximum depth will be automatically configured to handle this level of intermittent jitter without packet loss.

Actual jitter above the average but below the maximum may result in delayed audio play out while the jitter buffer adapts, but no packets will be lost. Actual jitter above the maximum value will always result in packet loss.

If legacy `voice.audioProfile.x.jitterBuffer.*` parameters are explicitly specified, they will be used to configure the jitter buffer and these `voice.rxQoS` parameters are ignored.

Change causes system to restart or reboot.

**voice.handsfreePtt.rxdg.offset**

This parameter allows a digital Rx boost for Push To Talk.

- 0 (default)
- 9 to -12 - Offsets the RxDg range of the hands-free and hands-free Push-to-Talk (PTT) by the specified number of decibels.

**voice.ringerPage.rxdg.offset**

This parameter allows a digital Rx boost for Push To Talk. Use this parameter for handsfree paging in high noise environments.

- 0 (default)
- 9 to -12 - Raise or lower the volume of the ringer and hands-free page by the specified number of decibels.

**Digital Gain Parameters**

Use the following parameters configure the gain applied to microphones.

**voice.handset.txdg**

Digital gain applied to the wired handset mic.

- 0 (Default)
- -90 to 90

**voice.handsfree.txdg**

Digital gain applied to the built-in hands free mic.

- 0 (Default)
- -90 to 90

**voice.headset.txdg**

Digital gain applied to the wired headset mic.

- 0 (Default)
- -90 to 90
**voice.usb.headset.txdg**

Digital gain applied to the USB headset mic.

- 0 (Default)
- -90 to 90

**voice.bt.headset.txdg**

Digital gain applied to the Bluetooth headset mic.

- 0 (Default)
- -90 to 90

### SDP Parameters

Use the following parameters to configure the Session Description Protocol (SPD).

**voIpProt.SDP.answer.useLocalPreferences**

- 0 (default) - Attempt to match the negotiated voice and video codecs using the order in the SDP offer from the far end.
- 1 - Answer SDP offers using the phone’s local preferences for codec ordering instead of the preference order from the offer.

**voIpProt.SDP.early.answerOrOffer**

- 0 (default) - SDP offer or answer is not generated.
- 1 - SDP offer or answer is generated in a provisional reliable response and PRACK request and response.

**Note:** An SDP offer or answer is not generated if `reg.x.musicOnHold.uri` is set.

**voIpProt.SDP.offer.iLBC.13_33kbps.includeMode**

- 1(default) - The phone should include the mode=30 FMTP parameter in SDP offers:
  - If you set `voice.codecPref.iLBC.13_33kbps`, and `voice.codecPref.iLBC.15_2kbps` is `null`.
  - If you set both `voice.codecPref.iLBC.13_33kbps` and `voice.codecPref.iLBC.15_2kbps`, the iLBC 13.33 Kbps codec is set to a higher preference.
- 0 - the phone should not include the mode=30 FTMP parameter in SDP offers even if iLBC 13.33 Kbps codec is being advertised.

**voIpProt.SDP.offer.rtcpVideoCodecControl**

This parameter determines whether or not RTCP-FB-based controls are offered in Session Description Protocol (SDP) when the phone negotiates video I-frame request methods. Even when RTCP-FB-based controls aren’t offered in SDP, the phone may still send and receive
RTCP-FB I-frame requests during calls depending on other parameter settings. For more information about video I-frame request behavior, see video.forceRtcpVideoCodecControl. For an account of all parameter dependencies refer to “I-Frames.”

section.

0 - The phone doesn't include the SDP attribute "a=rtcp-fb".
1 - The phone includes the SDP attribute "a=rtcp-fb" into offers during outbound SIP calls.

Download Location Parameter for Language Files

The following parameter specifies the download location of the translated language files for the system web interface (Web Configuration Utility).

webutility.language.plcmServerUrl

Specifies the download location of the translated language files for the system web interface.

http://downloads.polycom.com/voice/software/languages/

(default)

URL

XML Streaming Protocol Parameters

Use the following parameters to set the XML streaming protocols for instant messaging, presence, and contact lists for BroadSoft features.

xmpp.1.auth.domain

Specify the domain name of the XMPP server.

Null (Default)

Other values - UTF-8 encoded string

xmpp.1.auth.useLoginCredentials

Specifies whether or not to use the login credentials provided in the phone's Login Credentials menu for XMPP authentication.

0 (Default)

1

xmpp.1.enable

Specifies to enable or disable XMPP presence.

0 (Default)

1
Session Header Parameters

You can enable session header parameters to convey information between phones about the SIP message.

Use the following parameters to configure session header components.

**voIpProt.SIP.supportFor100rel**

1 (default) - The phone advertises support for reliable provisional responses in its offers and responses.

0 - The phone doesn't offer 100rel and rejects offers requiring 100rel.

**voIpProt.SIP.keepalive.sessionTimers**

0 (default) – The phone doesn't declare a timer in the Support header in an INVITE. The call doesn't disconnect when the phone doesn't receive UPDATE packet. The phone still responds to a re-INVITE or UPDATE and follows the session timer to send re-INVITE or UPDATE if the remote endpoint asks for it.

1 – The session timer is enabled and the call disconnects when the phone doesn't receive an UPDATE packet within the specified session timer.

**reg.x.keepalive.sessionTimers**

1 (default) – The session timer is enabled and the call received on the registered line disconnects when the phone doesn't receive an UPDATE packet within the specified timer.

0 – The session timer is disabled and the call received on the registered line doesn't disconnect when the phone doesn't receive an UPDATE packet.