



ADMINISTRATORS' GUIDE

Polycom UC Software 4.1.0 | October 2012 | 1725-11530-410 Rev B

Polycom UC Software 4.1.0

Administrators' Guide



Polycom UC Software 4.1.0 Administrators' Guide

Copyright ©2012, Polycom, Inc. All rights reserved.

Polycom, Inc.
6001 America Center Drive
San Jose, CA 95002
USA

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Polycom, Inc. Under the law, reproducing includes translating into another language or format.

As between the parties, Polycom, Inc., retains title to and ownership of all proprietary rights with respect to the software contained within its products. The software is protected by United States copyright laws and international treaty provision. Therefore, you must treat the software like any other copyrighted material (e.g., a book or sound recording).

Every effort has been made to ensure that the information in this manual is accurate. Polycom, Inc., is not responsible for printing or clerical errors. Information in this document is subject to change without notice.

Trademarks

POLYCOM®, the Polycom logo and the names and marks associated with Polycom products are trademarks and/or service marks of Polycom, Inc. and are registered and/or common law marks in the United States and various other countries. All other trademarks are property of their respective owners. No portion hereof may be reproduced or transmitted in any form or by any means, for any purpose other than the recipient's personal use, without the express written permission of Polycom.

Disclaimer

While Polycom uses reasonable efforts to include accurate and up-to-date information in this document, Polycom makes no warranties or representations as to its accuracy. Polycom assumes no liability or responsibility for any typographical or other errors or omissions in the content of this document.

Limitation of Liability

Polycom and/or its respective suppliers make no representations about the suitability of the information contained in this document for any purpose. Information is provided "as is" without warranty of any kind and is subject to change without notice. The entire risk arising out of its use remains with the recipient. In no event shall Polycom and/or its respective suppliers be liable for any direct, consequential, incidental, special, punitive or other damages whatsoever (including without limitation, damages for loss of business profits, business interruption, or loss of business information), even if Polycom has been advised of the possibility of such damages.

Customer Feedback

We are striving to improve our documentation quality and we appreciate your feedback. Email your opinions and comments to VoiceDocumentationFeedback@polycom.com.



Visit the [Polycom Voice Support Center](#) for software downloads, product documents, product licenses, troubleshooting tips, service requests, and more.

Contents

- About This Guide1**
 - Who Should Read This Guide? 1
 - How This Guide is Organized..... 1
 - What’s New in This Guide 3
 - Conventions Used in This Guide 3
 - Recommended Software Tools..... 5
 - Reading the Feature Parameter Tables 5
 - Example One: Feature Parameter Tables*..... 6
 - Example Two: Configuring Grouped Parameters* 7
 - Getting Help and Support 10

- Part I: Getting Started..... 13**
 - Chapter 1: Welcome to the Polycom® UC Software Family of Phones15**
 - The Polycom UC Software Family of Phones 15
 - Key Features of Your Polycom Phones..... 18
 - What’s New in Polycom UC Software 4.1.0?..... 20

 - Chapter 2: The Polycom® UC Software Big Picture23**
 - Where Polycom Phones Fit in Your Network..... 24
 - Understanding Polycom Phone Software Architecture 25
 - What is the Updater?*..... 26
 - What is the Polycom UC Software?*..... 27
 - What are the Configuration Files?*..... 28
 - What are the Resource Files?*..... 29

- Part II: Setting Up Your Environment..... 31**
 - Chapter 3: Setting Up Your Device Network.....33**
 - Establishing Link Connectivity..... 34
 - Wired Devices*..... 34
 - Wireless Devices*..... 34
 - Security and Quality of Service Settings 34
 - VLANs and Wired Devices* 34

802.1X Authentication	35
IP Communication Settings	36
PSTN Communication Settings	38
Provisioning Server Discovery	40
Supported Provisioning Protocols.....	41
Phone Network Menus	43
Main Menu.....	44
Provisioning Server Menu.....	45
DHCP Menu.....	47
Network Interfaces Menu (Ethernet Menu).....	48
VLAN Menu.....	50
802.1X Menu.....	51
PAC File Information.....	52
Wi-Fi Menu.....	52
WEP Menu.....	54
WPA (2) PSK Menu.....	54
WPA2-Enterprise Menu.....	55
Radio Menu.....	56
5 GHz Menu.....	56
2.4 GHz Menu.....	57
USBNet Menu.....	57
CMA Menu	58
Login Credentials Menu	58
TLS Menu.....	59
TLS Profile Menu.....	59
Applications Menu.....	60
Syslog Menu	60

Chapter 4: Setting Up the Provisioning Server63

Why Use a Provisioning Server?	64
Provisioning Server Security Notes.....	64
Setting up an FTP Server as Your Provisioning Server.....	65
Downloading Polycom UC Software Files to the Provisioning Server.....	66
Deploying and Updating Polycom Phones with a Provisioning Server.....	67
Deploying Polycom Phones with a Provisioning Server.....	68
Upgrading Polycom UC Software.....	70
Upgrading Current Phones to UC Software 4.1.0.....	71
Supporting Legacy Phones	73
Provisioning VVX 1500 Phones Using a Polycom CMA System.....	76
Provisioning Using Polycom CMA.....	77
Disabling the Polycom CMA System.....	78

<i>Upgrading Polycom UC Software Using Polycom CMA</i>	78
<i>Monitoring by Polycom CMA</i>	79
Provisioning SpectraLink 8400 Series Wireless Handsets	79

Chapter 5: Configuration Methods 81

Using the Centralized Provisioning Method - Configuration Files	83
<i>Understanding the Master Configuration File</i>	85
<i>Understanding Variable Substitution</i>	88
Example One	89
Example Two	89
<i>Using the Template Configuration Files</i>	90
<i>Changing Configuration Parameter Values</i>	92
Customizing Parameters for a Phone Model	93
Provisioning with the Web Configuration Utility	94
<i>Accessing the Web Configuration Utility</i>	95
<i>Choosing Language Files for the Web Configuration Utility Interface</i>	96
Phone User Interface – Menu System Settings	98

Part III: Configuring the Phone Features 101

Chapter 6: Setting Up Basic Phone Features 103

Basic Phone Features at a Glance	103
Configuring the Call Logs	105
<i>Example Call Log Configuration</i>	106
Understanding the Call Timer	107
Configuring Call Waiting Alerts	108
<i>Example Call Waiting Configuration</i>	108
Called Party Identification	109
Configuring Calling Party Identification	109
<i>Example Calling Party Configuration</i>	110
Configuring PSTN Calling Party Identification	111
Enabling Missed Call Notification	111
<i>Example Missed Call Notification Configuration</i>	112
Connected Party Identification	112
Distinctive Incoming Call Treatment	113
<i>Example Call Treatment Configuration</i>	113
Applying Distinctive Ringing	114
<i>Example Distinctive Ringing Configuration</i>	116
Applying Distinctive Call Waiting	116
<i>Example Distinctive Call Waiting Configuration</i>	117

Configuring Do Not Disturb.....	117
<i>Example Do Not Disturb Configuration.....</i>	<i>119</i>
Configuring the Handset, Headset, and Speakerphone	120
<i>Example Handset, Headset, and Speakerphone Configuration.....</i>	<i>121</i>
Using the Local Contact Directory	121
<i>Example Configuration.....</i>	<i>122</i>
Using the Local Digit Map	124
<i>Understanding Digit Map Rules.....</i>	<i>125</i>
Microphone Mute.....	127
Using the Speed Dial Feature	127
<i>Example Speed Dial Configuration.....</i>	<i>128</i>
Setting the Time and Date Display	130
<i>Example Configuration.....</i>	<i>130</i>
Adding an Idle Display Image.....	132
<i>Example Idle Display Image Configuration.....</i>	<i>133</i>
Ethernet Switch	134
Setting a Graphic Display Background	135
<i>Example Graphic Display Background Configuration.....</i>	<i>136</i>
Enabling Multikey Answer.....	138
<i>Example Multikey Answer Configuration.....</i>	<i>139</i>
Enabling Automatic Off-Hook Call Placement.....	139
<i>Example Automatic Off-Hook Placement Configuration.....</i>	<i>140</i>
Enabling Call Hold.....	140
<i>Example Call Hold Configuration.....</i>	<i>141</i>
Using Call Transfer	142
<i>Example Call Transfer Configuration.....</i>	<i>143</i>
Creating Local and Centralized Conferences.....	144
Enabling Conference Management.....	144
<i>Example Conference Management Configuration.....</i>	<i>145</i>
Configuring Call Forwarding.....	146
<i>Example Call Forwarding Configuration.....</i>	<i>147</i>
Configuring Directed Call Pick-Up.....	148
<i>Example Directed Call Pickup Configuration.....</i>	<i>149</i>
Enabling Group Call Pickup.....	150
<i>Example Group Call Pickup Configuration.....</i>	<i>151</i>
Configuring Call Park and Retrieve	152
<i>Example Call Park and Retrieve Configuration.....</i>	<i>153</i>
Enabling Last Call Return	154
<i>Example Configuration for Last Call Return.....</i>	<i>154</i>

Chapter 7: Setting Up Advanced Phone Features 157

Configuring the Phone's Keypad Interface.....	160
Assigning Multiple Line Keys Per Registration	161
<i>Example Configuration</i>	162
Enabling Multiple Call Appearances	162
<i>Example Multiple Call Appearances Configuration</i>	163
Customizing and Downloading Fonts	165
Setting the Phone Language.....	166
<i>Example Phone Language Configuration</i>	167
Enabling Instant Messaging	169
<i>Example Instant Messaging Configuration</i>	170
Synthesized Call Progress Tones	171
Using the Microbrowser and Web Browser.....	171
<i>Example Microbrowser and Web Browser Configuration</i>	173
Configuring Real-Time Transport Protocol Ports	175
<i>Example Real-Time Transport Protocol Configuration</i>	176
Configuring Network Address Translation.....	177
<i>Example Network Address Translation Configuration</i>	177
Using the Corporate Directory.....	178
<i>Example Corporate Directory Configuration</i>	179
CMA Directory	182
Recording and Playing Audio Calls	183
<i>Example Call Recording Configuration</i>	184
Configuring the Digital Picture Frame	186
<i>Example Digital Picture Frame Configuration</i>	187
Configuring Enhanced Feature Keys	188
<i>Some Guidelines for Configuring Enhanced Feature Keys</i>	189
<i>Enhanced Feature Key Examples</i>	190
<i>Understanding Macro Definitions</i>	192
<i>Macro Action</i>	192
<i>Prompt Macro Substitution</i>	193
<i>Expanded Macros</i>	194
<i>Special Characters</i>	194
<i>Example Macro</i>	194
Configuring Soft Keys.....	196
<i>Example Soft Key Configurations</i>	198
Enabling the Power Saving Feature.....	200
<i>Example Power-Saving Configuration</i>	201
Configuring Push-to-Talk and Group Paging.....	202
<i>Push-to-Talk</i>	202
<i>Group Paging</i>	203
<i>Example PTT/Paging Configuration</i>	205
PTT Mode Channels	206

Paging Mode Groups.....	207
Flexible Line Key Assignment.....	207
<i>Example Flexible Line Key Assignment Configuration.....</i>	<i>208</i>
Configuring Shared Call Appearances	209
<i>Example Configuration.....</i>	<i>211</i>
Phone A	212
Phone B	213
Enabling Bridged Line Appearance.....	213
<i>Example Bridged Line Appearance Configuration.....</i>	<i>214</i>
Using Busy Lamp Field	215
<i>Example BLF Configuration.....</i>	<i>217</i>
Enabling Voicemail Integration.....	220
<i>Example Voicemail Configuration.....</i>	<i>221</i>
Enabling Multiple Registrations.....	222
<i>Example Multiple Registration Configuration.....</i>	<i>223</i>
Using Hoteling	226
<i>Example Hoteling Configuration</i>	<i>226</i>
Configuring SIP-B Automatic Call Distribution.....	227
<i>Example SIP-B Automatic Call Distribution Configuration.....</i>	<i>229</i>
Configuring Feature-Synchronized Automatic Call Distribution (ACD).....	231
<i>Example Feature Synchronized ACD Configuration</i>	<i>233</i>
Setting Up Server Redundancy	236
DNS SIP Server Name Resolution.....	238
<i>Behavior When the Primary Server Connection Fails.....</i>	<i>238</i>
Phone Configuration	239
Phone Operation for Registration	239
<i>Recommended Practices for Fallback Deployments.....</i>	<i>240</i>
Using the Presence Feature.....	240
<i>Example Presence Configuration.....</i>	<i>241</i>
Using CMA Presence	243
Enabling Access URL in SIP Messages.....	244
<i>Example Access URL in SIP Messages Configuration.....</i>	<i>246</i>
Configuring the Static DNS Cache	247
<i>Example Static DNS Cache Configuration.....</i>	<i>248</i>
Displaying SIP Header Warnings	251
<i>Example Display of Warnings from SIP Headers Configuration.....</i>	<i>251</i>
Quick Setup of Polycom Phones	252
<i>Example Quick Setup Configuration.....</i>	<i>253</i>
Provisional Polling of Polycom Phones	254
<i>Example Provisional Polling Configuration</i>	<i>255</i>
Setting Up Microsoft Office Communications Server 2007 R2 Integration	256
<i>Example OCS 2007 R2 Integration Configuration (Single Registration).....</i>	<i>256</i>

Setting Up Microsoft Lync Server 2010 Integration	261
<i>Registering with Microsoft Lync Server 2010</i>	262
<i>Example Configuration: Setting the Base Profile to Lync</i>	263
Enabling Polycom Desktop Connector Integration	266
<i>Example PDC Configuration</i>	267
Enabling Microsoft Exchange Calendar Integration	268
<i>Example Exchange Calendar Configuration</i>	269
Configuring the Polycom Quick Barcode Connector Application	271
<i>Example QBC Configuration</i>	271
Configuring the Open Application Interface	273
<i>Example OAI Configuration</i>	273
Enabling Location Services	274
<i>Example Location Service Integration Configuration</i>	274
Chapter 8: Setting Up Phone Audio Features	275
Customizing Audio Sound Effects	276
<i>Example Configuration</i>	277
Context Sensitive Volume Control	278
Voice Activity Detection	279
Generating Dual Tone Multi-Frequency (DTMF) Tones	279
DTMF Event RTP Payload	280
Acoustic Echo Cancellation	280
Audio Codecs	281
IP Type-of-Service	284
IEEE 802.1p/Q	285
Voice Quality Monitoring	285
Audible Ringer Location	287
Notification Profiles	287
Bluetooth Headset Support	288
Built-In Audio Processing Features	288
<i>Automatic Gain Control</i>	288
<i>Background Noise Suppression</i>	289
<i>Comfort Noise Fill</i>	289
<i>Dynamic Noise Reduction</i>	289
<i>Jitter Buffer and Packet Error Concealment</i>	289
<i>Low-Delay Audio Packet Transmission</i>	289
Chapter 9: Setting Up Phone Video Features	291
Video Transmission	291
Video Codecs	293
H.323 Protocol	294

<i>Supported Video Standards</i>	295
<i>Supported Polycom Interoperability</i>	296
<i>Using the H.323 Protocol</i>	297
Switching Between Voice and Video During Calls.....	299

Chapter 10: Setting Up User and Phone Security Features 301

Local User and Administrator Passwords	302
Incoming Signaling Validation	303
Configuration File Encryption	303
Digital Certificates	304
Generating a Certificate Signing Request.....	306
TLS Profiles.....	307
<i>Downloading Certificates to a Polycom Phone</i>	309
<i>Configuring TLS Profiles</i>	310
Supporting Mutual TLS Authentication	310
Configurable TLS Cipher Suites	312
Secure Real-Time Transport Protocol	313
Locking the Phone	315
Locking the Keypad on Your SpectraLink Handset.....	317
Secondary Port Link Status Report.....	317
Supporting 802.1X Authentication	318
Using User Profiles	320
Creating a Phone Configuration File	322
Creating a User Configuration File	323

Part IV: Troubleshooting and Maintaining your Deployment 325

Chapter 11: Troubleshooting Your Polycom Phones..... 327

Understanding Error Message Types	328
<i>Updater Error Messages</i>	328
<i>Polycom UC Software Error Messages</i>	328
Status Menu	332
Log Files	332
<i>Reading a Boot Log File</i>	335
<i>Reading an Application Log File</i>	337
<i>Reading a Syslog File</i>	338
Managing the Phone's Memory Resources	338
<i>Identifying Symptoms</i>	339
<i>Checking the Phone's Available Memory</i>	339

Managing the Phone Features..... 340

Testing Phone Hardware 342

Uploading a Phone’s Configuration..... 342

Network Diagnostics 343

Ports Used on Polycom Phones 343

Power and Startup Issues..... 344

Dial Pad Issues..... 345

Screen and System Access Issues 345

Calling Issues..... 346

Display Issues..... 347

Audio Issues..... 348

Licensed Feature Issues 348

Upgrading Issues 349

SoundStation Duo Failover Issues 351

Chapter 12: Miscellaneous Maintenance Tasks 353

Trusted Certificate Authority List..... 353

Encrypting Configuration Files 356

Polycom UC Software Dependencies..... 358

Supported VVX 1500 and CMA Server Interoperability 359

Multiple Key Combinations 359

Rebooting the Phone..... 360

Resetting to Factory Defaults..... 360

Updating Log Files..... 361

Setting Base Profile 361

Default Feature Key Layouts..... 362

Internal Key Functions..... 374

Assigning a VLAN ID Using DHCP 378

Parsing Vendor ID Information..... 380

Product, Model, and Part Number Mapping..... 381

Disabling the PC Ethernet Port..... 383

Capturing the Phone’s Current Screen..... 383

LLDP and Supported TLVs..... 384

Supported TLVs..... 386

 System and Model Names..... 389

 PMD Advertise and Operational MAU..... 390

 Power Values 390

Part V: Polycom® UC Software Reference Information 393

Chapter 13: Configuration Parameters395

<acd/>	398
<apps/>	398
<attendant>	402
<bg/>	404
<bitmap/>	407
<bluetooth/>	407
<button/>	407
<call/>	408
<callLists/>	414
<device/>	414
<dialplan/>	427
<dir>	431
<local/>	431
<corp/>	431
<divert/>	434
<dns/>	435
DNS-A	435
DNS-NAPTR	436
DNS-SRV	437
<efk/>	438
<exchange/>	440
<feature/>	440
	443
<hoteling/>	444
<httpd/>	445
<key/>	445
<keypadLock/>	447
<lcl/>	448
<ml/>	448
<datetime/>	450
<i>SoundStation Duo Localization Preferences</i>	451
<license/>	453
<lineKey/>	453
<loc/>	453
<log/>	455
<level/> <change/>and<render/>	456
<sched/>	457
<mb/>	458
<messaging/>	459
<msg/>	460

<nat/>	461
<np/>	461
<oai/>	475
<phoneLock/>	476
<powerSaving/>.....	477
<pres/>	478
<prov/>	479
<pstn/>	481
<ptt/>	482
<qbc/>	486
<qos/>	486
<reg/>	488
<request/>	498
<roaming_buddies/>	499
<roaming_privacy/>	499
<saf/>	500
<se/>	501
<pat/>.....	502
<rt/>.....	506
<sec/>	508
<encryption/>	508
<pwd/><length/>.....	509
<srtp/>.....	510
<H235/>.....	512
<dot1x><eapollogoff/>.....	513
<hostmovedetect/>.....	513
<TLS/>	513
<profile/>	515
<profileSelection/>.....	516
<softkey/>	517
<tcpIpApp/>	519
<dhcp/>.....	519
<dns/>	520
<ice/>.....	520
<snmp/>	521
<port/><rtp/>	523
<keepalive/>.....	524
<fileTransfer/>.....	525
<tones/>	525
<DTMF/>.....	525
<chord/>.....	526
<up/>	527

<upgrade/>	532
<video/>	532
<codecs/>.....	534
<codecPref/>	534
<profile/>	535
<camera/>	538
<localCameraView/>.....	539
<voice/>	539
<codecPref/>.....	540
<volume/>.....	542
<vad/>	543
<quality monitoring/>	543
<rxQoS/>	545
<volpProt/>	546
<server/>.....	546
<SDP/>	550
<SIP/>	551
<H323/>.....	559
<webutility/>	560
<Wi-Fi/>	561

Chapter 14: Session Initiation Protocol (SIP) 563

RFC and Internet Draft Support.....	563
<i>Request Support</i>	565
<i>Header Support</i>	566
<i>Response Support</i>	569
1xx Responses - Provisional	569
2xx Responses - Success.....	570
3xx Responses - Redirection.....	570
4xx Responses - Request Failure.....	570
5xx Responses - Server Failure.....	572
6xx Responses - Global Failure	572
<i>Hold Implementation</i>	572
<i>Reliability of Provisional Responses</i>	573
<i>Transfer</i>	573
<i>Third Party Call Control</i>	573
<i>SIP for Instant Messaging and Presence Leveraging Extensions</i>	574
<i>Shared Call Appearance Signaling</i>	574
<i>Bridged Line Appearance Signaling</i>	574

Chapter 15: Polycom UC Software Menu System 575

Chapter 16: Third-Party Software581

About This Guide

The Polycom® UC Software Administrator's Guide provides instructions for installing, provisioning, and administering Polycom phones. This guide will help you understand the Polycom VoIP network and telephony components, and provides descriptions of all available phone features. Specifically, this Administrators' Guide will help you perform the following tasks:

- Install and configure your phone on a network server or Web server
- Configure your phone's features and functions
- Configure your phone's user settings
- Troubleshoot common phone issues

Who Should Read This Guide?

System administrators and network engineers should read this guide to learn how properly to set up Polycom phones. This guide describes administration-level tasks and is not intended for end users.

Before reading this guide, you should be familiar with the following:

- Computer networking and driver administration for your operating system
- An XML editor
- The XML-based configuration file format that the Polycom UC Software and its supported phones use

How This Guide is Organized

This guide is organized into six parts, each containing multiple chapters. The parts and their chapters are sequenced in the order you deploy Polycom phones.

- *Part I: Getting Started* gives you an overview of the Polycom phones and Polycom Unified Communications (UC) Software.
- *Part II: Setting Up Your Environment* gives you information on setting up your phone network, a provisioning server, and how to use the configuration methods.
- *Part III: Configuring the Phone Features* is devoted to descriptions of the phone features you can configure and brief example configurations.
- *Part IV: Troubleshooting and Maintaining Your Deployment* identifies common phone issues and includes troubleshooting tips, and explains a number of software and hardware maintenance tasks.

- *Part V: Polycom UC Software Reference Information* is a complete account of the parameters you can set to configure phone features, and includes a description, permissible values, and the default value. There are chapters you can use to understand the Session Initiation Protocol (SIP) and the phone menu system, and you can view copyright statements for third-part software products that run on Polycom phones.

The following is a list of the parts and chapters included in each part:

Part I: Getting Started

Chapter 1: Welcome to the Polycom UC Software Family of Phones introduces the Polycom phones that support the latest Polycom UC Software.

Chapter 2: The Polycom UC Software Big Picture shows you how Polycom phones fit in your organization and details about the Polycom UC Software architecture.

Part II: Setting Up Your Environment

Chapter 3: Setting Up Your Phone Network describes how to set up your network.

Chapter 4: Setting Up the Provisioning Server provides basic and advanced instructions on how to set up a provisioning server, deploy the Polycom phones from the provisioning server, and upgrade the phone's software.

Chapter 5: Configuration Methods explains the three configuration methods you can use to configure the phone features and settings.

Part III: Configuring the Phone Features

Chapter 6: Setting Up Basic Phone Features explains how to configure and use basic phone features like call waiting and speed dials.

Chapter 7: Setting Up Advanced Phone Features shows you how to configure and use advanced phone features like corporate directory and voice mail.

Chapter 8: Setting Up Audio Features provides information on configuring and using audio features like voice quality monitoring.

Chapter 9: Setting Up Phone Video Features shows you how to configure and use video features like the H.323 protocol.

Chapter 10: Setting Up User and Phone Security Features describes how to configure and use security features like locking the phone.

Part IV: Troubleshooting and Maintaining Your Deployment

Chapter 11: Troubleshooting Your Polycom Phones explains error messages and how to read the phone's log files.

Chapter 12: Miscellaneous Maintenance Tasks gives information about tasks like displaying a logo on a phone display and taking pictures of the phone's screen.

Part V: Polycom UC Software Reference Information

Chapter 13: Configuration Parameters provides detailed descriptions of all of the configuration parameters that the Polycom UC Software uses.

Chapter 14: Session Initiation Protocol (SIP) provides information on the SIP RFCs supported by the Polycom UC Software.

Chapter 15: Polycom UC Software Menu System shows the menu structure of the Polycom UC Software as it displays on Polycom phones.

Chapter 16: Third Party Software outlines licensing information on the third party software used by the Polycom UC Software.

What's New in This Guide

The content in this guide has been significantly revised for clarity and to provide more information to system administrators who are new to deploying Polycom phones. Specifically, *Chapter 5: Configuration Methods* has been created to centralize information on the configuration methods you can use to configure phone features and settings. *Part III: Configuring the Phone Features* has been modified to include clearer descriptions and example configurations.

Conventions Used in This Guide

The following icons are used to alert you to various types of important information in this guide:

Icons Used in this Guide

Name	Icon	Description
Note		The <i>Note</i> icon highlights information of interest or important information needed to be successful in accomplishing a procedure or to understand a concept.
Administrator Tip		The <i>Administrator Tip</i> icon highlights techniques, shortcuts, or productivity related tips.
Caution		The <i>Caution</i> icon highlights information you need to know to avoid a hazard that could potentially impact device performance, application functionality, or successful feature configuration.
Warning		The <i>Warning</i> icon highlights an action you must perform (or avoid) to prevent issues that may cause you to lose information or your configuration setup, and/or affect phone or network performance.

<i>Name</i>	<i>Icon</i>	<i>Description</i>
Web Info		The <i>Web Info</i> icon highlights supplementary information available online such as documents or downloads on support.polycom.com or other locations.
Timesaver		The <i>Timesaver</i> icon highlights a faster or alternative method for accomplishing a method or operation.
Power Tip		The <i>Power Tip</i> icon highlights faster, alternative procedures for advanced administrators already familiar with the techniques being discussed.
Troubleshooting		The <i>Troubleshooting</i> icon highlights information that may help you solve a relevant problem or to refer you to other relevant troubleshooting resources.
Settings		The <i>Settings</i> icon highlights settings you may need to choose for a specific behavior, to enable a specific feature, or to access customization options.

A few typographic conventions, listed next, are used in this guide to distinguish types of in-text information.

Typographic Conventions

<i>Convention</i>	<i>Description</i>
Bold	Highlights interface items such as menus, soft keys, file names, and directories. Also used to represent menu selections and text entry to the phone.
<i>Italics</i>	Used to emphasize text, to show example values or inputs, and to show titles of reference documents available from the Polycom Support Web site and other reference sites.
Blue	Used for cross-references to other sections, chapters, or parts in this document.
<u>Underlined Blue</u>	Used for URL links to external Web pages or documents. If you click on text in this style, you will be linked to an external document or Web page.
Blue Text	Used for cross references to other sections within this document. If you click on text in this style, you will be taken to another part of this document.
Fixed-width-font	Used for code fragments and parameter names.

This guide also uses a few writing conventions to distinguish conditional information.

Writing Conventions

Convention	Description
<MACaddress>	Indicates that you must enter information specific to your installation, phone, or network. For example, when you see <MACaddress>, enter your phone's 12-digit MAC address. If you see <installed-directory>, enter the path to your installation directory.
>	Indicates that you need to select an item from a menu. For example, Settings > Basic indicates that you need to select Basic from the Settings menu.
parameter.*	Used for configuration parameters. If you see a parameter name in the form parameter.* , the text is referring to all parameters beginning with parameter. See Reading the Feature Parameter Tables for an example.

Recommended Software Tools

Polycom recommends that you use an XML editor – such as XML Notepad – to create and edit configuration files. In this way, all configuration files that you create will be valid XML files.

If the configuration files are not valid XML, they will not load on the handset and an error message will be logged to the provisioning server.

Reading the Feature Parameter Tables

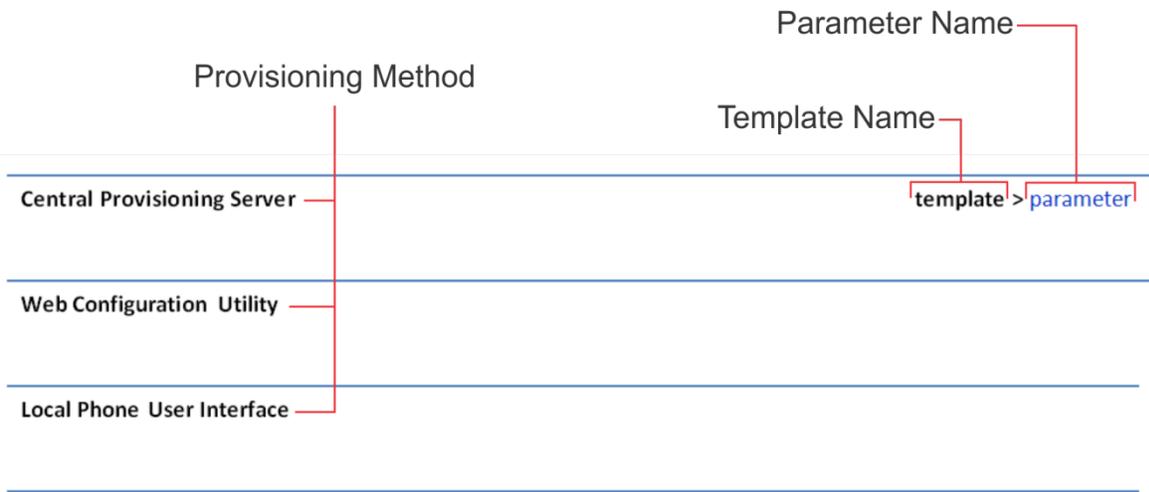
Each of the feature descriptions discussed in *Part III: Configuring the Phone Features* includes a table of parameters that you configure to make the features work. This brief section explains the conventions used in the feature parameter tables. Polycom strongly recommends gaining familiarity with these conventions in order to read the tables and successfully perform configuration changes.

The feature parameter tables, as shown in Figure 1, indicate one or more of three provisioning methods you can use to configure a feature: a centralized provisioning server, the Web Configuration Utility, or the local phone user interface. Note that the types of provisioning methods available for each feature will vary; not every feature uses all three methods.

The central provisioning server method requires you to configure parameters located in template configuration files that Polycom provides in XML format. Figure 1 shows you how to

use the parameter tables to locate the template name and the name of the parameter you configure to get the phone features working.

Feature Parameter Table Format



To quickly locate a specific parameter, locate and open the template name indicated. Then, use the parameter name to navigate the folders in the XML tree structure. The parameter name contains the XML folder path. The two following examples explain this convention in more detail.

Example One: Feature Parameter Tables

The example table shown in Figure 2 is taken from *Setting the Time and Date Display* in Chapter 5.

Feature Parameter Table for Time and Date Display

Central Provisioning Server	template > parameter
Turn the time and date display on or off.....	reg-advanced.cfg > up.localClockEnabled

Figure 2 indicates that the **reg-advanced.cfg** template file contains the `up.localClockEnabled` parameter, which turns the time and date display on or off. This parameter is enabled by default. If you want to turn the time and date display on or off, locate and open the **reg.advanced** template, expand the **up** folder, and locate the parameter name `up.localClockEnabled`. Set the parameter value to '1' to turn on or '0' to turn off the time and date display, as shown in the following illustration.

Example Time and Date Display

Template Name

Folder Name

Parameter Name

Parameter Value

Note that some of the file paths in the templates are long and you may have to expand several folders in the XML tree structure to locate a specific parameter.

Note also that some feature parameters are located in more than one template file. In these cases, the parameter tables will list all related template files.



Tip: Each Parameter Is Linked

Each parameter listed in the tables in Chapter 5, 6, 7, 8, and 9 is linked to its definition in Chapter 14. The sections in Chapter 14 define each parameter and list the permissible values, including the default value, of each parameter. If you want to find out more about a parameter you see listed in the tables, click on the parameter.

Example Two: Configuring Grouped Parameters

Some of the features have several related parameters that you will need to configure to get the feature working. In these cases, instead of listing every parameter, the table will specify a group of related parameters with an abbreviated XML path name ending with (*.*) , which indicates you can configure a group of related parameters.

Abbreviated XML paths, like full parameter names, are linked to their definitions in the reference sections in Chapter 14. Specifically, since the reference sections lists parameters alphabetically, abbreviated XML path are linked to the first of a group of parameters listed alphabetically in the reference section. Figure 4, shown next, shows you that in the **site.cfg** template, the `tcpIpApp.sntp` folder contains several related parameters that configure basic SNTP settings.

Feature Parameter Table for Time and Date SNTP Settings

Central Provisioning Server	template > parameter
Set the basic SNTP settings and daylight savings parameters.....	site.cfg > tcpIpApp.sntp.*

Figure 4 indicates that there is a group of SNTP parameters you can configure in the **site.cfg** template file. The abbreviated parameter name `tcpIpApp.sntp.*` indicates that you can configure parameters in the `tcpIpApp.sntp` folder as well as parameters in `tcpIpApp.sntp` subfolders.

To locate these parameters in the XML file, use the parameter name. The parameter name contains the XML folder path, as shown in the following illustration.

Locating Parameters in the Templates

Template Name

The screenshot shows the XML Notepad interface with the file `C:\Poly650\site.cfg` open. The tree view on the left displays the following structure:

- xml
 - #comment
 - polycomConfig
 - xmlns:xsi
 - xsi:noNamespaceSchemaLocation
 - call
 - tcpIpApp
 - tcpIpApp.dns
 - tcpIpApp.fileTransfer
 - tcpIpApp.keepalive
 - tcpIpApp.port
 - tcpIpApp.sntp
 - tcpIpApp.sntp.address
 - tcpIpApp.sntp.gmtOffset
 - tcpIpApp.sntp.resyncPeriod
 - tcpIpApp.sntp.address
 - tcpIpApp.sntp.address.overrideDHCP
 - tcpIpApp.sntp.daylightSavings
 - tcpIpApp.sntp.daylightSavings.enable
 - tcpIpApp.sntp.daylightSavings.fixedDayEnable
 - tcpIpApp.sntp.daylightSavings.start
 - tcpIpApp.sntp.daylightSavings.start.date
 - tcpIpApp.sntp.daylightSavings.start.dayOfWeek
 - tcpIpApp.sntp.daylightSavings.start.month
 - tcpIpApp.sntp.daylightSavings.start.time
 - tcpIpApp.sntp.daylightSavings.start.dayOfWeek
 - tcpIpApp.sntp.daylightSavings.start.dayOfWeek.lastInMonth
 - tcpIpApp.sntp.daylightSavings.stop
 - tcpIpApp.sntp.daylightSavings.stop.date
 - tcpIpApp.sntp.daylightSavings.stop.dayOfWeek
 - tcpIpApp.sntp.daylightSavings.stop.month
 - tcpIpApp.sntp.daylightSavings.stop.time
 - tcpIpApp.sntp.daylightSavings.stop.dayOfWeek
 - tcpIpApp.sntp.daylightSavings.stop.dayOfWeek.lastInMonth
 - tcpIpApp.sntp.gmtOffset
 - tcpIpApp.sntp.gmtOffset.overrideDHCP

The right pane shows the XML code with values:

```

version="1.0"
Generated site
http://www.w3.
polycomConfig.

0
86400
0
0
1
0
8
1
3
2
0
1
1
11
2
0
0

```

In cases where the feature has several related parameters, you may find it helpful to refer to the parameter reference section in *Chapter 13* for a definition of each parameter. All parameter names, including abbreviated names, are linked to the parameter reference section - simply click on the parameter name.

This section has shown you how to read the configuration parameter tables so that you can locate the parameters in the XML template file.



Tip: Using an XML Editor

Polycom recommends using an XML editor such as XML Notepad 2007 to open and edit the configuration template files.

Getting Help and Support

If you are looking for help or technical support for your phones, the following types of documents are available at the [Polycom Support Center](#):

- Quick Start Guides, which describe how to assemble phones
- Quick User Guides, which describe the basic phone features
- User Guides, which describe both basic and advanced phone features
- Web Applications Developer's Guide, which provides guidance in the development of applications that run on your phone's Web browser or microbrowser
- Feature Description and Technical Notifications such as Technical Bulletins and Quick Tips that describe workarounds to existing issues and provide expanded descriptions and examples
- Release Notes, which describe the new and changed features and fixed problems in the latest version of the software

You can find Request for Comments (RFC) documents by entering the RFC number at <http://www.ietf.org/rfc.html>.

For other references, look for the Web Info icon  throughout this Administrators' Guide.



For support or service, please contact your Polycom reseller or visit support.polycom.com for software downloads, product documents, product licenses, troubleshooting tips, service requests, and more.

We are constantly working to improve the quality of our documentation, and we would appreciate your feedback. Please send email to VoiceDocumentationFeedback@polycom.com.

Polycom recommends that you record the phone model numbers, software (both the Updater and UC Software), and partner platform for future reference.

Phone models: _____

Updater version: _____

UC Software version: _____

Partner Platform: _____

Part I: Getting Started

Part I gives you an overview of the Polycom® phones and of the Polycom UC Software and consists of the following chapters:

- [Chapter 1: Welcome to the Polycom UC Software Family of Phones](#)
- [Chapter 2: The Polycom UC Software Big Picture](#)

Chapter 1: Welcome to the Polycom[®] UC Software Family of Phones

This chapter introduces the family of Polycom[®] phones that support Polycom UC Software version 4.1.

The Polycom family of phones provides a powerful, yet flexible IP communications solution for Ethernet TCP/IP networks. Not only do the phones deliver excellent voice quality, but also come with a high-resolution graphic display screen for call information, multiple languages, directory access, and system status. The phones can also support advanced functionality, including multiple call and flexible line appearances, HTTPS secure provisioning, presence, custom ringtones, and local conferencing.

From an administrator's perspective, the phones are endpoints in an overall network topology designed to interoperate with other compatible equipment including application servers, media servers, internet-working gateways, voice bridges, and other end points.

The following models are described:

- [SoundPoint IP Desktop Phones](#)
- [SoundStation IP Conference Phones](#)
- [V VX Business Media Phones](#)
- [SpectraLink 8400 Series Wireless Handsets](#)
- [SoundStructure VoIP Interface](#)

The Polycom UC Software Family of Phones

This section provides you with a graphic list of the Polycom family of phones that support UC Software 4.1.



Web Info: Support for Polycom Phones

You can find all documentation for all Polycom phones on the [Polycom Support](#) site. Choose your phone model for specific documentation. For more information, contact your Polycom distributor.

Table 1-1: The Polycom Family of Phones

SoundPoint IP Desktop Phones

Polycom currently supports the following desktop phones:

SoundPoint IP 321, 331, and 335



SoundPoint IP 450



SoundPoint IP 550 and 560



SoundPoint IP 650



SoundStation IP Conference Phones

Polycom currently supports the following conference phones:

SoundStation IP 5000



SoundStation IP 6000



SoundStation Duo



VVX Business Media Phones

Polycom currently supports the following business media phones:

VVX 500



VVX 1500



SpectraLink 8400 Series Wireless Handsets

Polycom currently supports the following wireless handsets:

SpectraLink 8440



SpectraLink 8450



SoundStructure VoIP Interface

The SoundStructure VoIP Interface is a plug-in card for SoundStructure products that adds SIP telephony to any SoundStructure system. SoundStructure products are used for installed-room audio and video conferencing applications.

SoundStructure VoIP Interface



Key Features of Your Polycom Phones

Polycom phones running Polycom UC Software include the following key features:

- Award winning sound quality with a full-duplex speakerphone or conference phone
 - Permits natural, high-quality, two-way conversations
 - Uses Polycom industry leading Acoustic Clarity Technology
 - Most phone models support Polycom HDVoice™ Technology

- Easy-to-use
 - An easy transition from traditional PBX systems into the world of IP Communications
 - Up to 18 dedicated hard keys for access to commonly used features
 - Up to four context-sensitive soft keys for further menu-driven activities
- Platform independent
 - Supports multiple protocols and platforms enabling standardization of one phone for multiple locations, systems, and vendors
- Faster Boot Time
 - The time between phone reboot and obtaining a dial tone has been noticeably reduced.
- Field upgradeable
 - Upgrade phones as standards develop and protocols evolve
 - Extends the life of the phone to protect your investment
 - Application flexibility for call management and new telephony applications
- Large LCD
 - Easy-to-use, easily readable, and intuitive interface
 - Support of rich application content, including multiple call appearances, presence and instant messaging, and XML services
 - 102 x 23 pixel graphical LCD for the SoundPoint IP 321/331/335
 - 256 x 116 pixel graphical grayscale LCD for the SoundPoint IP 450 (supports Asian characters)
 - 320 x 160 pixel graphical grayscale LCD for the SoundPoint IP 550/560/650 (supports Asian characters)
 - 248 x 68 pixel graphical LCD for the SoundStation IP 5000
 - 800 x 480 pixel graphical color LCD for the VVX 1500 (touch screen)
 - 240 x 320 pixel graphical color LCD for the SpectraLink handsets
- Dual auto-sensing 10/100/1000baseT Ethernet ports on certain Polycom phones
 - Leverages existing infrastructure investment
 - No re-wiring with existing CAT 5 cabling
 - Simplifies installation
 - 1000baseT is supported by the SoundPoint IP 560, VVX 1500, and the SoundStructure VoIP Interface.
- Power over Ethernet (PoE) port or Power Pack option
 - Built-in IEEE 802.3af PoE port on the SoundPoint IP 320/321/330/331/335, 450, 550, 560, and 650, the SoundStation IP 5000 and 6000, and VVX 500 and 1500 (auto-sensing)

- Unused pairs on Ethernet port are used to deliver power to the phone via a wall adapter, meaning fewer wires on your desktop (for the SoundStation IP 6000 conference phones)
- Multiple language support on most phones
 - Set on-screen language to your preference. Select from Chinese (Simplified and Traditional), Danish, Dutch, English (Canada, United Kingdom, and United States), French, German, Italian, Japanese, Korean, Norwegian, Polish, Portuguese (Brazilian), Russian, Slovenian, Spanish (International), and Swedish.

Note that Japanese and Korean are not supported on the SoundPoint IP 321, 331, or 335 phones.
- Web Browser
 - Supports a subset of XHTML constructs that run like any other Web browser
- Browser on the Polycom VVX 500 and 1500 phones and SpectraLink handsets
 - Supports XHTML 1.1 constructs, HTML 4.01, JavaScript, CCS 2.1, and SVG 1.1 (partial support)
- XML status/control API
 - Ability to poll phones for call status and device information
 - Ability to receive telephony notification events

For more information, see the *Web Applications Developer's Guide* available from [Polycom Support](#).

What's New in Polycom UC Software 4.1.0?

The following changes were introduced in UCS 4.1.0:

- [Setting Up Microsoft Lync Server 2010 Integration](#)
- Support for the VVX 500 Business Media Phone
- Hoteling enhancement to feature-synchronized Automatic Call Distribution (ACD)
- Additional symbology on SpectraLink 8452 Series wireless handsets



Note: Polycom UC Software 4.1.0 does not support the SoundStation IP 6000 phone.

UC Software 4.1.x does not support the SoundStation IP 6000 phone. Because the next future UC Software release will support the IP 6000, references to the IP 6000 have been left in this UC Software 4.1.0 Administrators' Guide.



Note: SoundPoint IP 670 and SoundStation IP 7000 phones are no longer supported

The SoundPoint IP 670 and SoundStation IP 7000 phones are not supported beyond UC Software 4.0.x and are not supported by UC Software 4.1.x. These phones are now legacy phones and can be updated with the UC Software 4.0.x releases.

Chapter 2: The Polycom® UC Software Big Picture

This chapter provides an overview of the Polycom® UC Software, specifically an understanding of how the phones fit into the network configuration. If you want to begin setting up your Polycom phones, go to [Setting Up Your Phone Network](#).

The UC Software supports the deployment of Polycom phones in several deployment scenarios:

- As a Session Initiation Protocol (SIP)-based endpoint interoperating with a SIP call server or softswitch.
- As an H.323 video endpoint (Polycom® VVX® 1500 business media phones only).



Web Info: Using VVX 1500 Phones in a Strict H.323 Environment

For more information on using VVX 1500 phones in a strict H.323 environment, see the [Deployment Guide for the Polycom VVX 1500 D Business Media Phone](#).

- In an 802.1X wireless environment (Polycom® SpectraLink® 8400 Series Wireless Phone).



Web Info: Using SpectraLink Handsets in a Strictly Wireless Environment

For more information on using these handsets in a strictly wireless environment, see the [Polycom SpectraLink 8400 Series Wireless Telephone Deployment Guide](#).

The Session Initiation Protocol (SIP) is the Internet Engineering Task Force (IETF) standard for multimedia communications over IP. It is an ASCII-based, application-layer control protocol (defined in RFC 3261) that can be used to establish, maintain, and terminate calls between two or more endpoints. Like other voice over IP (VoIP) protocols, SIP is designed to address the functions of signaling and session management within a packet telephony network. Signaling allows call information to be carried across network boundaries. Session management provides the ability to control the attributes of an end-to-end call.

For Polycom phones to successfully operate as a SIP endpoint in your network, you will require:

- A working IP network
- Routers configured for VoIP
- VoIP gateways configured for SIP

- The latest (or a compatible version) Polycom UC Software image
- An active, configured call server to receive and send SIP messages

For information on IP PBX and softswitch vendors, see the [Polycom ARENA VoIP Interoperability Partners list](#).

The rest of this chapter consists of the following sections:

- [Where Polycom Phones Fit in Your Network](#)
- [Understanding Polycom Phone Software Architecture](#)

If you want to begin setting up your Polycom phones on the network, go to [Setting Up Your Phone Network](#).

If you want to begin configuring the features available for your Polycom phones, go to [Part III: Configuring the Phone Features](#).

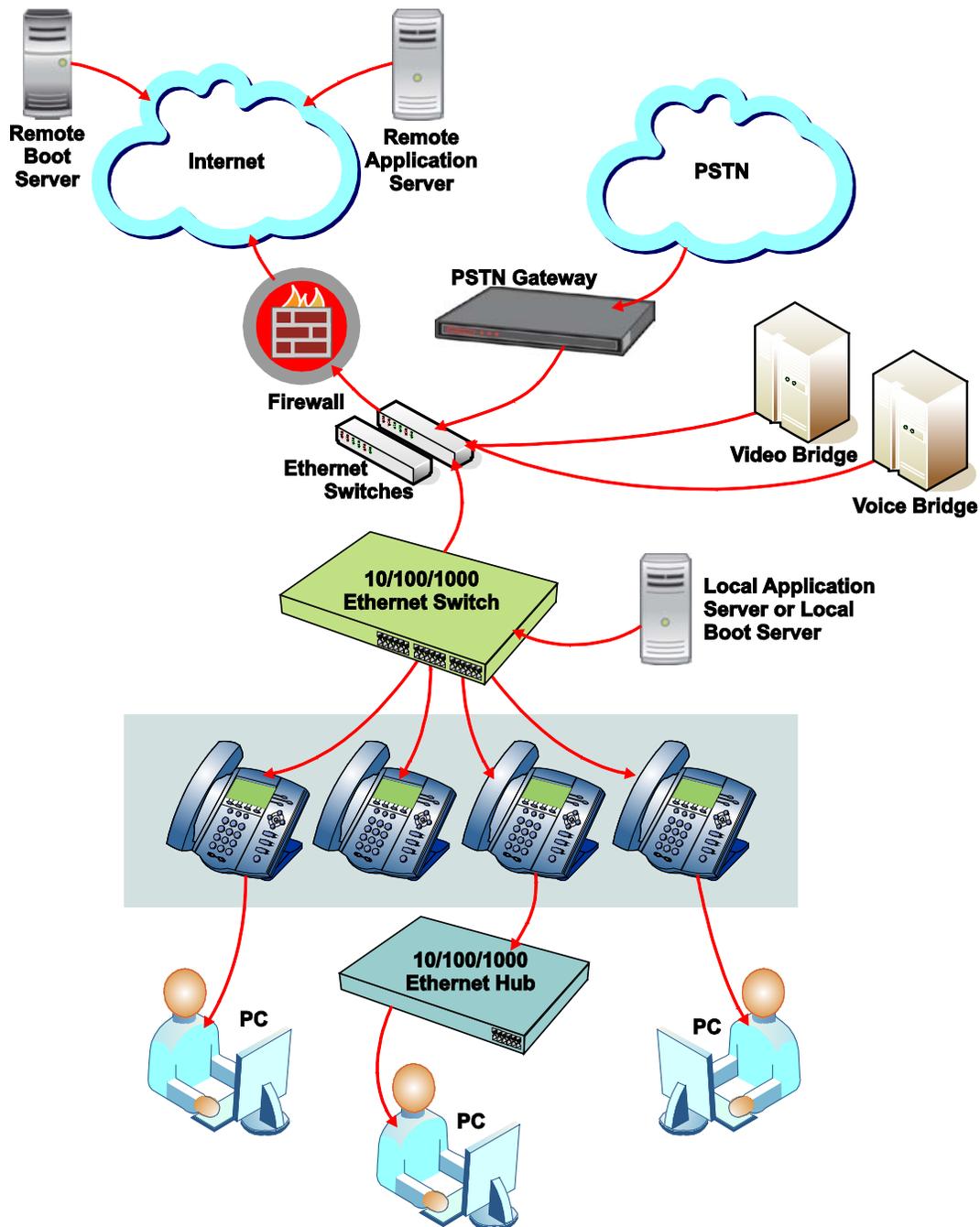
Where Polycom Phones Fit in Your Network

Most Polycom phones connect physically through a Category 5 (Cat-5) cable to a standard office twisted-pair (IEEE 802.3) 10/100/1000 megabits per second Ethernet LAN, and send and receive all data using the same packet-based technology. SpectraLink wireless handsets connect to a WLAN. [Figure 2-1: Polycom Wired Phones in a Network](#) shows wired phones in a network.

Since the phone is a data terminal, digitized audio being just another type of data from its perspective, the phone is capable of vastly more than traditional business phones. Moreover, Polycom phones run the same protocols as your office personal computer, which means that many innovative applications can be developed without resorting to specialized technology.

There are many ways to set up a phone network using Polycom phones and [Figure 2-1](#), shown next, is just one example of a network setup.

Figure 2-1: Polycom Wired Phones in a Network



Understanding Polycom Phone Software Architecture

The Polycom phone software is made of four basic components:

- **Updater** The software that loads first when the phone is powered on
- **Polycom UC Software** The software that implements the phone functions and features

- **Configuration files** The files that contain the phone's settings
- **Resource files** Optional files that contain settings for advanced features

Figure 2-2: Polycom Phone Software Architecture



What is the Updater?

The Updater is a small application that resides in the flash memory on the phone. Polycom phones, except the SoundStation IP 6000, come installed with the Updater.



Note: The Updater is also Known as BootROM

The Updater was referred to as the BootROM in previous versions of the UC Software, specifically UC Software 3.3.x and SIP 3.2.x and earlier.

When you start/boot/reboot the phone, the Updater performs the following tasks:

- 1 Enables you to open the setup menu so you can set various network and provisioning options.

The Updater requests IP settings and accesses the provisioning server (also called the boot server) to look for any changes to the Updater software.

If updates are found, they are downloaded and saved to flash memory, which overwrites itself after verifying the integrity of the download.

- 2 If new updates are downloaded, the Updater formats the file system, removes any application software and configuration files that were present.
- 3 Downloads the master configuration file.
The Updater and the application use this file to acquire a list of other files that the phone needs.
- 4 Examines the master configuration file for the name of the application file, and then looks for this file on the provisioning server.
If the copy on the provisioning server is different than the one stored in device settings, or there is no file stored in flash memory, the application file is downloaded.
- 5 Extracts the Polycom UC software from flash memory.
- 6 Installs the application into RAM, then uploads an event log file from the boot cycle.

The Updater will then terminate, and the Polycom UC software will take over.

What is the Polycom UC Software?

The Polycom Unified Communications Software, or Polycom UC Software, manages the protocol stack, the digital signal processor (DSP), the user interface, and the network interaction. The UC Software implements the following functions and features on the phones:

- VoIP signaling for a wide range of voice and video telephony functions using SIP signaling for call setup and control
- SIP and H.323 signaling for video telephony
- Industry standard security techniques for ensuring that all provisioning, signaling, and media transactions are robustly authenticated and encrypted
- Advanced audio signal processing for handset, headset, and speakerphone communications using a wide range of audio codecs
- Flexible provisioning methods to support single phone, small business, and large multi-site enterprise deployments

The software is a binary file image and contains a digital signature that prevents tampering or the loading of rogue software images.

There is a new image file in each release of software.

Both the Updater and Polycom UC Software run on all phone models that Polycom currently supports. For a list of unsupported phone models, see [Supporting Legacy Phones](#).

Current build archives have both split and combined images. You can decide which model(s) to support. Using split files reduces internal network traffic during reboots and updates. Using combined files means you need to download only one software file.

What are the Configuration Files?

The Polycom UC Software that you download contains template configuration files, valid XML files that you can change using an XML editor. These template files contain a number of parameters that provision the phones with features and settings. The template configuration files are very flexible: you can rearrange the parameters within the template, move parameters to new files, or create your own configuration files from only those parameters you want. This flexibility is useful when you want to apply the same features and settings to a large number of phones. Use of the configuration files to provision the phones with features and settings is called the centralized provision method – the configuration files enable you to store a single set of configuration files on a central provisioning server and configure all of your phones to read the same set of files.

Polycom recommends that you configure phones using the centralized provisioning method. However, there are several methods you can use to configure the phones and you can use one or multiple methods in combination. If you are using a Polycom VVX 1500 business media phone, you can use the Polycom Converged Management Application™ (CMA™) server. Alternatively, you can configure individual phones using the phone's menu system, accessible through the local user interface, or using Web Configuration Utility.

You will need to keep in mind that there is a hierarchy among the configuration methods and settings you make using a higher-priority method will override settings you make using a lower-priority method. The following lists all of the available ways to set features and settings for the phones. Each of these methods is detailed in [Chapter 5: Configuration Methods](#). Polycom strongly recommends becoming familiar with each of the configuration methods.

Configuration Methods

You can make changes to the phone's configuration using any of the following configuration methods. Take note that there is a precedence order among the configuration methods: changes made to settings using a higher-priority method override settings made using a lower-priority method. Configuration changes are uploaded to the phone as override files that remain active until you remove them or reset to default.

The precedence order for configuration parameter changes is as follows (highest to lowest priority):

- Local phone user interface
- Web Configuration Utility
- Polycom CMA system
- Central Provisioning Server
- Default values

Each of these configuration methods is detailed in [Chapter 5: Configuration Methods](#).

What are the Resource Files?

In addition to the software and configuration files, the phones may require resource files in order to use some of the advanced features.

Examples of resource files include:

- Language dictionaries
- Custom fonts
- Ringtones
- Contact directories

If you need to remove resource files from a phone at a later date - for example, if you are giving the phone to a new user - you will have to apply factory default settings to that phone.



Web Info: Resetting Your Phone to the Factory Default Settings

For instructions on how to reset your phone to factory default settings, see [Quick Tip 18298: Updating, Troubleshooting, and Resetting SoundPoint IP, SoundStation IP, and VVX 1500 Phones](#).

Part II: Setting Up Your Environment

Part II provides you with essential information on how to set up your phone network and provisioning server, and on the configuration methods you can use to set up phone features. You will find basic and advanced instructions on how to set up a provisioning server, how to deploy the Polycom® phones from the provisioning server, and how to upgrade the software.

Part II consists of the following chapters:

- [Chapter 3: Setting Up Your Phone Network](#)
- [Chapter 4: Setting Up the Provisioning Server](#)
- [Chapter 5: Configuration Methods](#)

Chapter 3: Setting Up Your Device Network

Polycom® SoundPoint® IP, VVX®, and SoundStation® phones using Polycom UC Software operate on an Ethernet local area network (LAN). The SpectraLink 8400 Series Wireless Handsets operate on a Wi-Fi LAN (WLAN). The SoundStation Duo™ can operate on a LAN or a Public Services Telephony Network (PSTN). Local area network design varies by organization and Polycom phones can be configured to accommodate a number of network designs. This chapter shows you several automated and manual ways to configure Polycom phones to operate in a LAN.

Connecting your Polycom phone to the LAN will initiate a startup sequence. Note that only step 1 is required and automatic (except for phones on a WLAN). Steps 2, 3, and 4 are optional as all these settings can be manually configured on the device. It is common to complete step 3 using a DHCP server within the LAN. The phone uses the following startup sequence:

- 1 The phone establishes network connectivity.
Wired phones will establish a 10M/100M/1000M network link with an Ethernet switch device. Wireless handsets will establish a Wi-Fi (802.11a/b/g/n) connection to a wireless access point. Neither phone will function until this link is established. If the phone cannot establish a link to the LAN, an error message *Link is Down* will display.
- 2 Apply appropriate security and Quality of Service (QoS) settings (optional).
Assign the phone to a VLAN and/or 802.1X authentication.
- 3 Establish DHCP negotiation with the network and IP address, network addressing options, network gateway address, and time server.
- 4 Provision server discovery.
This is commonly done using DHCP as part of the previous step.

Once the provisioning server discovery is complete the phone will initiate the provisioning process described in [Chapter 4: Setting Up the Provisioning Server](#).

These steps are described in more detail in the following sections of this chapter:

- [Establishing Link Connectivity](#)
- [Security and Quality of Service Settings](#)
- [IP Communication Settings](#)
- [PSTN Communication Settings](#)
- [Phone Network Menus](#)

Establishing Link Connectivity

Wired and wireless devices will establish a connection to the LAN or WLAN, respectively. If you want to change the phone's configuration, do so prior to connecting the devices.

Wired Devices

Typical network equipment supports one of the three following Ethernet line rates: 10Mbps, 100Mbps, and 1000Mbps. The phones are configured to automatically negotiate the Ethernet rate so that no special configuration is required. You do have the option to change the line rates and/or duplex configuration. Polycom recommends that you keep the default settings. If you do change the settings, you should do so before deploying the phones.

The phone supports two features to prevent Denial of Service (DoS):

- **Storm Filtering** To change this parameter, go to [Network Interfaces Menu \(Ethernet Menu\)](#).
- **VLAN Filtering** To change this parameter, go to [VLAN Menu](#). Note that VLAN filtering is not supported on the VVX family of phones.

Wireless Devices

You must configure wireless devices before they can establish a connection to a wireless network. You can configure wireless devices manually, but it is more common to configure them prior to deployment using the USB interface (USBNet) to the device (and the `device.set` parameters in the configuration file).

To change the wireless settings that may need to be set up to connect your device to the Wireless LAN (WLAN), go to [Wi-Fi Menu](#).

Security and Quality of Service Settings

You have the option of using several layer-2 mechanisms that increase network security and minimize audio latency. This section describes each of the network security options.

VLANs and Wired Devices

A Virtual LAN (VLAN) can be used to separate and assign higher priority to a voice VLAN as a way of minimizing latency.

There are several methods in which the phone can be configured to work on a particular VLAN:

- **LLDP** Link Layer Discovery Protocol (LLDP) is a vendor-neutral Layer 2 protocol that allows a network device to advertise its identity and capabilities on the local network. To change these parameters, go to [VLAN Menu](#).

- **CDP Compatible** Cisco Discovery Protocol (CDP) is a proprietary Data Link Layer network protocol . CDP Compatible follows the same set of rules. To change this parameter, go to [VLAN Menu](#).
- **DHCP** Dynamic Host Configuration Protocol (DHCP) is an automatic configuration protocol used on IP networks. To change this parameter, go to [DHCP Menu](#). To use DHCP for assigning VLANs, see [Assigning a VLAN ID Using DHCP](#). Note that the use of DHCP for assigning VLANs is not well standardized and is recommended only if the switch equipment does not support LLDP or CDP Compatible methods.
- **Static** The VLAN ID can be manually set from the phone UI or from a configuration file. To change this parameter, go to [VLAN Menu](#). This will set the device setting parameter only.

If the phone receives a VLAN setting from multiple of the above methods, the priority is as follows (from highest to lowest):

- LLDP
- CDP
- Device settings
- DHCP VLAN discovery

802.1X Authentication

802.1X authentication is a technology that originated for authenticating Wi-Fi links. It has also been adopted for authenticating PCs within fixed LAN deployments.

When VoIP phones (with a secondary Ethernet port) are used to connect PCs on a network the 802.1X authentication process becomes more complex since the PC is not directly connected to the 802.1X switch.



Web Info: 802.1X References

For more information on 802.1X authentication, see Introduction to IEEE 802.1X and Cisco® Identity-Based Networking Services (IBNS) at http://www.cisco.com/en/US/products/ps6662/products_ios_protocol_option_home.html

See also [IEEE 802.1X Multi-Domain Authentication on Cisco Catalyst Layer 3 Fixed Configuration Switches Configuration Example](#).

There are several ways to configure 802.1X authentication of devices connected to the PC port of the phone:

- You can configure many switches to automatically *trust* or accept a VoIP phone based on its MAC address. This is sometimes referred to as MAC Address Bypass (MAB).

- Some switches support a feature whereby they will to automatically *trust* a device that requests a VLAN using the CDP protocol.
- Some deployments support Multiple Device Authentication (MDA). In this situation, both the phone and the PC will separately authenticate themselves.

In this scenario since the phone is closest to the 802.1X switch, the phone needs to notify the switch when the PC is disconnected. This can be achieved using an 802.1X EAPOL-Logoff message.

All three of the above are supported by Polycom products. Multiple Device Authentication is new in UC Software 4.0.0.

To change these parameters, go to [802.1X Menu](#).

IP Communication Settings

When the phone has established network connectivity it needs to acquire several IP network settings to proceed with provisioning. These settings are typically obtained automatically from a DHCP server.



Tip: Tip For Novice Administrators

Read this section if you are new to this process or have never set up a provisioning server before.

You have the option to set the IP communication settings manually from the phone UI, or to pre-provision using a `device.set` capability.

When making the DHCP request the phone will include information in Option 60 that can assist the DHCP server in delivering the appropriate settings to the device. For more information, see [Technical Bulletin 54041: Using DHCP Vendor Identifying Options With Polycom Phones](#).



Timesaver: Reducing Repetitive Data Entry

Polycom recommends using DHCP where possible to eliminate repetitive manual data entry.

The following table details the settings that are supported through the DHCP menu:

Table 3-1: DHCP Network Parameters

| <i>Parameter</i> | <i>DHCP Option</i> | <i>DHCP</i> | <i>DHCP
INFORM</i> | <i>Configuration File
(application only)</i> | <i>Device
Settings</i> |
|------------------------------|---|---|------------------------|--|----------------------------|
| IP address | - | • | - | - | • |
| Subnet mask | 1 | • | - | - | • |
| IP gateway | 3 | • | - | - | • |
| Boot server address | See DHCP Menu or Provisioning Server Discovery . | • | • | - | • |
| SIP server address | 151 Note: You can change this value by changing the device setting. See <device/> | • | - | - | • |
| SNTP server address | Look at option 42, then option 4. | • | - | • | • |
| SNTP GMT offset | 2 | • | - | • | • |
| DNS server IP address | 6 | • | - | - | • |
| DNS INFORM server IP address | 6 | • | - | - | • |
| DNS domain | 15 | • | - | - | • |
| VLAN ID | See DHCP Menu . | Warning: Link Layer Discovery Protocol (LLDP) overrides Cisco Discovery Protocol (CDP). CDP overrides Local FLASH which overrides DHCP VLAN Discovery. | | | |



Web Info: RFC Information on DHCP Options

For more information on DHCP options, see [RFC 2131](#) and [RFC 2132](#).



Note: Overriding the DHCP Value

The configuration file value for **SNTP server address** and **SNTP GMT offset** can be configured to override the DHCP value. See [tcpIpApp.sntp.address.overrideDHCP](#). The CDP Compatibility value can be obtained from a connected Ethernet switch if the switch supports CDP.

If you do not have control of your DHCP server or do not have the ability to set the DHCP options, you will need to enable the phone to automatically discover the provisioning server address. One way is to connect to a secondary DHCP server that responds to DHCP INFORM queries with a requested provisioning server value. For more information, see [RFC 3361](#) and [RFC 3925](#).

PSTN Communication Settings

The SoundStation Duo conference phone is the only Polycom phone running Polycom UC Software that supports PSTN mode. The PSTN communication settings described in this section apply only to SoundStation Duo conference phones.

The SoundStation Duo has several connection modes, each with its own default behavior. You can change both the connection and the default behavior of a connection method.

There are three ways to connect your Polycom SoundStation Duo conference phone:

- To an analog phone jack only.
By default, the phone will only operate in Public Switched Telephone Network (PSTN) mode.
- To a Session Internet Protocol (SIP) call server only.
By default, the phone will only operate in SIP mode.
- To both a SIP server and an analog phone jack.
By default, the phone will automatically operate in SIP mode. If the phone is unable to register with a SIP server, or the network connection is unavailable, the phone will operate in PSTN mode.

You can override the default behavior and specify one of three operational modes the phone will use.

- Auto (Automatic Mode Detect)
This is the default operational mode. Select this option if you want your phone to automatically select which operational mode it uses. The phone will use PSTN mode if it is unable to register with a SIP server or if the network connection is unavailable.
- PSTN Only
Select this option if you want your phone to operate exclusively in PSTN mode.
- SIP Only

Select this option if you want your phone to operate exclusively in SIP mode.



Note: Changing Modes During a Call

If you change operational modes during a call, it will have no effect on the current call. Your next call will use the new mode.

After you connect the phone for PSTN use, you need to configure two basic settings for the phone to operate properly in PSTN mode:

- The language the phone will use.
- The country in which the phone is located.

The phone requires this basic information to automatically configure other PSTN settings, since several PSTN settings are country specific.



Web Info: Changing Basic Settings on the SoundStation Duo

For more information on setting up the phone for use with PSTN, see the SoundStation Duo Quick Start Guide at [Polycom Support](#).

After you configure the country and language, you can configure more advanced PSTN settings, such as the PSTN extension, caller ID, caller ID type, and flash timing, by navigating to the **Advanced** menu on the phone. The following table lists the advanced PSTN settings you can configure.

Table 3-2: Advanced PSTN Settings

| <i>Name</i> | <i>Permitted Values</i> | <i>Default</i> |
|---------------------|-------------------------------|---|
| Flash Timing | 80, 100, 300, 600 (ms) | The default value depends on the country code that is configured for the phone. The flash duration is automatically set to the default for that country. |

The length of time before a hook flash times-out (or the call disconnects). The flash duration is based on the country of origin that is specified for the phone.

| <i>Name</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|-----------------|
| Caller ID | On, Off, Removed | On |
| <p>Caller ID displays a caller's phone number (and possibly a name), on the called party's phone. Specify whether caller ID is on, off, or removed. If caller ID is removed, the Caller ID Type menu item is removed from the phone's menu.</p> <p>Note: Caller ID is a subscription service. Check with your local telephone service provider to determine if this service is available in your area.</p> <p>For additional information about Caller ID, see Configuring PSTN Calling Party Identification.</p> | | |
| PSTN Extension | Numerical string, up to a maximum of 32 numbers | Null |
| <p>The phone's telephone number. The number will display on the idle screen.</p> | | |
| Caller ID Type | Bellcore
ETSI

DTMF
Off | Bellcore |
| <p>The type of caller ID to use. If the value for Caller ID is 'Removed,' Caller ID Type won't display in the phone's menu. Note: The British Telecom and Japanese Caller ID standard is not supported. If you're using the phone in the United Kingdom or Japan, choose the 'Removed' option in the Caller ID parameter.</p> | | |

Provisioning Server Discovery

After the phone has established network connectivity it proceeds to the *Configuration* stage. In this stage the following steps are carried out:

- Software Update
- Application of configuration settings relevant to a customer network



Tip: Novice Administrators

Read this section if you are new to this process or have never set up a provisioning server before.

In many deployments a centralized provisioning server is used for the software update and configuration functions. The phone supports several methods to 'discover' this provisioning server:

- **Static** You can manually configure the server address from the phone's user interface or the Web Configuration Utility, or you can pre-provision the phone. The server address is manually configured from the phone's user interface, the Web Configuration Utility, or pre-provisioned using `device.set` in a configuration file.
- **DHCP** A DHCP option is used to provide the address or URL between the provisioning server and the phone.
- **DHCP INFORM** The phone makes an explicit request for a DHCP option (which can be answered by a server that is not the primary DHCP server). For more information, see [RFC 3361](#) and [RFC 3925](#).
- **Quick Setup** This feature offers a soft key to the user that takes them directly to a screen to enter the provisioning server address and information. This is simpler than navigating the menus to the relevant places to configure the provisioning parameters. For more information, see [Technical Bulletin 45460: Using Quick Setup with Polycom Phones](#).

To change these parameters, go to [Provisioning Server Menu](#).



Web Info: Provisioning Polycom Phones

For more information on best practices with respect to provisioning, see [White Paper 60806: UC Software Provisioning Best Practices](#).

Supported Provisioning Protocols

The Updater performs the provisioning functions of uploading log files, master configuration files, software updates, and device setting menu changes.

By default, phones are shipped with FTP enabled as the provisioning protocol. You can change the provisioning protocol by updating the *Server Type* option. Or, you can specify a transfer protocol in the *Server Address*, for example, `http://usr:pwd@server` (see [Provisioning Server Menu](#)). The *Server Address* can be an IP address, domain string name, or URL. It can be obtained through DHCP.

Configuration file names in the `<MACAddress>.cfg` file can include a transfer protocol, for example, `https://usr:pwd@server/dir/file.cfg`. If a user name and password are specified as part of the server address or file name, they will be used only if the server supports them. If a user name and password are required but not specified, the device settings are sent to the server.



Tip: Choosing a Valid URL

A URL should contain forward slashes instead of back slashes and should not contain spaces. Escape characters are not supported. If a user name and password are not specified, the *Server User* and *Server Password* from device settings will be used (see [Provisioning Server Menu](#)).



Note: Active and Passive FTP Methods

There are two types of FTP methods - active and passive. UC Software is not compatible with active FTP.



Note: HTTP/HTTPS Authentication

Both digest and basic authentication are supported when using HTTP/HTTPS for UC Software. Only digest authentication is supported when using HTTP by the Updater.

To guarantee software integrity, the Updater will download only cryptographically signed Updater or UC Software images. For HTTPS, widely recognized certificate authorities are trusted by the phone and custom certificates can be added to the phone.



Web Info: To View Trusted Certificate Authorities

For more information, see [Trusted Certificate Authority List](#) and [Technical Bulletin 17877: Using Custom Certificates With Polycom Phones](#).

As of SIP 3.2, Mutual Transport Layer Security (TLS) authentication is available. For more information, see [Supporting Mutual TLS Authentication](#).

As of UC Software 4.0.0, 802.1X authentication is available. For more information, see [Supporting 802.1X Authentication](#).

Digest Authentication for Microsoft Internet Information Services (IIS)

If you want to use digest authentication against the Microsoft Internet Information Services server:

- Use Microsoft Internet Information Server 6.0 or later.
- Digest authentication needs the user name and password to be saved in reversible encryption.
- The user account on the server must have administrative privileges.
- The wildcard must be set as MIME type; otherwise, the phone will not download *.cfg, *.ld and other required files. This is because the Microsoft Internet Information Server cannot recognize these extensions and will return a "File not found" error. To configure wildcard for MIME type, see <http://support.microsoft.com/kb/326965>.

For more information, see <http://www.microsoft.com/technet/prodtechnol/WindowsServer2003/Library/IIS/809552a3-3473-48a7-9683-c6df0cdfda21.msp?mfr=true>.

Phone Network Menus

You have the option of modifying the phone network configuration.



Tip: Tip For Novice Administrators

Read this section if you are new to this process or have never set up a provisioning server before.

You can update the network configuration parameters in two ways:

- **During the Updater Phase.** The setup menu is accessible during the auto-boot countdown of the Updater phase of operation. While your phone boots up, press the **Cancel** soft key, and press the **Setup** soft key to launch the setup menu. To access the setup menu, you will have to enter the administrator's password.
- **After your phone starts and is running UC Software.** The network configuration menu is accessible from the phone's main menu. Select **Menu > Settings > Advanced > Admin Settings > Network Configuration**. To access the **Advanced** menu, you will have to enter the administrator's password.



Tip: Changing the Default Administrator Password

Polycom recommends that you change the default administrative password. See [Local User and Administrator Passwords](#).

You have the option of modifying the phone network configuration parameters in the following menus and sub-menus:

- [Main Menu](#)
- [Provisioning Server Menu](#)
- [Network Interfaces Menu \(Ethernet Menu\)](#)
- [CMA Menu](#)
- [TLS Menu](#)
- [Syslog Menu](#)

Use the soft keys, the arrow keys, and the Select and Delete keys to make changes.

Certain parameters are read-only due to the value of other parameters. For example, if the **DHCP** client parameter is enabled, the **Phone IP Address** and **Subnet Mask** parameters are grayed out or not visible since the DHCP server automatically supplies these parameters and the statically assigned IP address and subnet mask will never be used in this configuration.



Tip: Resetting Network Configurations

The basic network configuration referred to in the subsequent sections can be reset to factory default settings using the phone's main menu: Select **Menu > Settings > Advanced > Admin Settings > Reset to Defaults > Reset Device Settings**. Or use a multiple key combination, as described in [Multiple Key Combinations](#).

Main Menu

You can modify the following configuration parameters from the setup menu while the phone boots, or from the Administrative Settings menu from a phone running UC Software:

Table 3-3: Main Menu

| <i>Name</i> | <i>Possible Values</i> |
|---|--|
| Provisioning Menu | |
| See Provisioning Server Menu . | |
| Network Interfaces Menu or Ethernet Menu | |
| See Network Interfaces Menu (Ethernet Menu) . | |
| CMA Menu | |
| See CMA Menu . | |
| TLS Menu | |
| See TLS Menu . | |
| SNTP Address | Dotted-decimal IP address OR Domain name string |
| The Simple Network Time Protocol (SNTP) server the phone obtains the current time from. | |
| GMT Offset | -13 through +12 |
| The offset of the local time zone from Greenwich Mean Time (GMT) in half hour increments. | |
| DNS Server | Dotted-decimal IP address |
| The primary server the phone directs Domain Name System (DNS) queries to. | |
| DNS INFORM Server | Dotted-decimal IP address |
| The secondary server to which the phone directs DNS queries. | |
| DNS Domain | Domain name string |
| The phone's DNS domain. | |

| <i>Name</i> | <i>Possible Values</i> |
|---|--------------------------|
| Hostname | hostname |
| The DHCP client hostname. | |
| Syslog Menu | |
| See Syslog Menu . | |
| Quick Setup | Enabled, Disabled |
| If enabled, a QSetup soft key displays. When you press the soft key, a menu displays that lets you configure the parameters required to access the provisioning server. | |
| Note: The Quick Setup option is not available in the Updater. | |
| EM Power | Enabled, Disabled |
| SoundPoint IP 650 only. This parameter is relevant if the phone uses a Power over Ethernet (PoE) source. If enabled, the phone will set power requirements in CDP to an appropriate value so that up to three Expansion Modules (EMs) can be powered. If disabled, the phone will set power requirements in CDP to an appropriate value which means no EMs will be powered, and no EMs will work. For exact power requirements, see Technical Bulletin 48152: Power Consumption of Polycom Phones . | |



Settings: Preventing Invalid Parameter Values

A parameter value of ??? indicates that the parameter has been set to an invalid value and saved in the phone's configuration. Before you complete your configuration, make sure you set values for these parameters.



Tip: Text Entry on the SoundPoint IP 321, 331, and 335 Phones

To switch the text entry mode on SoundPoint IP 321, 331, and 335 phones, press the # key. You may want to use URL or IP address modes when entering server addresses.

Provisioning Server Menu

The following configuration parameters can be modified on the Provisioning Server Menu:

Table 3-4: Provisioning Server Menu

| <i>Name</i> | <i>Possible Values</i> |
|--|------------------------|
| DHCP Menu | |
| See DHCP Menu . Note: This menu is disabled when the DHCP client is disabled. | |

| Name | Possible Values |
|---|---|
| Server Type | 0=FTP, 1=TFTP, 2=HTTP, 3=HTTPS, 4=FTPS |
| <p>The protocol that the phone will use to obtain configuration and phone application files from the provisioning server. See Supported Provisioning Protocols.</p> <p>Note: Active FTP is not supported for BootROM version 3.0 or later. Passive FTP is supported. Only implicit FTPS is supported.</p> | |
| Server Address | Dotted-decimal IP address OR URL |
| <p>Domain name string or a URL. All addresses can be followed by an optional directory. The address can also be followed by the file name of a .cfg master configuration file, which the phone will use instead of the default <MACaddress>.cfg file. The provisioning server to use if the DHCP client is disabled, if the DHCP server does not send a boot server option, or if the Boot Server parameter is set to Static. The phone can contact multiple IP addresses per DNS name. These redundant provisioning servers must all use the same protocol. If a URL is used, it can include a user name and password. See Supported Provisioning Protocols. For information on how to specify a directory and use the master configuration file, see Understanding the Master Configuration File.</p> <p>Note: ":", "@", or "/" can be used in the user name or password if they are correctly escaped using the method specified in RFC 1738.</p> | |
| Server User | String |
| <p>The user name requested when the phone logs into the server (if required) for the selected Server Type.</p> <p>Note: If the <i>Server Address</i> is a URL with a user name, this will be ignored.</p> | |
| Server Password | String |
| <p>The password requested when the phone logs in to the server if required for the selected Server Type.</p> <p>Note: If the <i>Server Address</i> is a URL with user name and password, this will be ignored.</p> | |
| File Transmit Tries | 1 to 10 Default 3 |
| <p>The maximum number of attempts to transfer a file. (An attempt is defined as trying to download the file from all IP addresses that map to a particular domain name.)</p> | |
| Retry Wait | 0 to 300 seconds Default 1 |
| <p>The minimum amount of time that must elapse before retrying a file transfer. The time is measured from the start of a transfer attempt, which is defined as the set of upload/download transactions made with the IP addresses that map to a given provisioning server's DNS. If the set of transactions in an attempt is equal to or greater than the Retry Wait value, then there will be no further delay before the next attempt is started.</p> <p>For more information, see Deploying Phones from the Provisioning Server.</p> | |
| Tag SN to UA | Disabled, Enabled |
| <p>If enabled, the phone's serial number (MAC address) is included in the User-Agent header of HTTP/HTTPS transfers and communications to the browser.</p> <p>The default value is Disabled.</p> | |

| Name | Possible Values |
|---|--------------------------|
| Upgrade Server | String |
| The address/URL that will be accessed for software updates requested from the phones Web configuration utility. | |
| ZTP | Disabled, Enabled |
| ZTP is a solution that Polycom plans to offer at a future date. | |



Tip: Changing the Default Passwords

The Server User and Server Password parameters should be changed from the default values.

DHCP Menu

The DHCP menu is accessible only when the DHCP client is enabled. You can update the following DHCP configuration parameters from the DHCP menu:

Table 3-5: DHCP Menu

| Name | Possible Values |
|---|--|
| Boot Server | 0=Option 66, 1=Custom, 2=Static, 3=Custom+Option 66 |
| <p>Option 66: The phone will look 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 <i>Server Address</i> in Provisioning Server Menu.</p> <p>Custom: The phone will look for the option number specified by the <i>Boot Server Option</i> parameter (below), and the type specified by the <i>Boot Server Option Type</i> parameter (below) in the response received from the DHCP server.</p> <p>Static: The phone will use the boot server configured through the <i>Server Menu</i>. For more information, see Provisioning Server Menu.</p> <p>Custom + Option 66: The phone will use the custom option first or use Option 66 if the custom option is not present.</p> <p><i>Note:</i> If the DHCP server sends nothing, the following scenarios are possible:</p> <ul style="list-style-type: none"> • If a boot server value is stored in flash memory and the value is not 0 . 0 . 0 . 0, then the value stored in flash is used. • Otherwise the phone sends out a DHCP INFORM query. <ul style="list-style-type: none"> ○ If a single DHCP INFORM server responds, this is functionally equivalent to the scenario where the primary DHCP server responds with a valid boot server value. ○ If no DHCP INFORM server responds, the INFORM query process will retry and eventually time out. • If the server address is not discovered using DHCP INFORM then the phone will contact the ZTP server if the ZTP feature is enabled. | |

| Name | Possible Values |
|---|---|
| Boot Server Option | 128 through 254 (Cannot be the same as VLAN ID Option) |
| When the <i>Boot Server</i> parameter is set to Custom, this parameter specifies the DHCP option number in which the phone will look for its boot server. | |
| Boot Server Option Type | 0=IP Address, 1=String |
| When the <i>Boot Server</i> parameter is set to Custom, this parameter specifies the type of DHCP option in which the phone will look for its provisioning server. The IP Address provided must specify the format of the provisioning server. The String provided must match one of the formats described for <i>Server Address</i> in Provisioning Server Menu . | |
| Option 60 Format | 0=RFC 3925 Binary, 1=ASCII String |
| <p>RFC 3925 Binary: Vendor-identifying information in the format defined in RFC 3925.</p> <p>ASCII String: Vendor-identifying information in ASCII.</p> <p>For more information, see Technical Bulletin 54041: Using DHCP Vendor Identifying Options With Polycom Phones.</p> <p>Note: DHCP option 125 containing the RFC 3295 formatted data will be sent whenever option 60 is sent. DHCP option 43 data is ignored.</p> | |



Note: Multiple DHCP INFORM Servers

If multiple DHCP INFORM servers respond, the phone should gather the responses from these DHCP INFORM servers. If configured for Custom+Option66, the phone will select the first response that contains a valid *custom* option value. If none of the responses contain a *custom* option value, the phone will select the first response that contains a valid *option66* value.

Network Interfaces Menu (Ethernet Menu)

The Network Interfaces Menu appears only if there are multiple network interfaces to the phone. For supported SoundPoint IP, SoundStation IP, and VVX phones, the Ethernet menu will display instead of the Network Interfaces menu. For SpectraLink handsets, the Network Interfaces menu will display.

You can select the following items in the Network Interfaces menu:

- Ethernet Menu (see [Table 3-6: Ethernet Menu](#))
- [Wi-Fi Menu](#) (SpectraLink handsets only)
- [USBNet Menu](#) (SpectraLink handsets only)

You can select the following items in the Ethernet menu:

Table 3-6: Ethernet Menu

| <i>Name</i> | <i>Possible Values</i> |
|---|--|
| DHCP | Enabled, Disabled |
| If enabled, DHCP will be used to obtain the parameters discussed in IP Communication Settings. | |
| IP Address | Dotted-decimal IP address |
| The phone's IP address.
Note: This parameter is disabled when DHCP is enabled. | |
| Subnet Mask | Dotted-decimal subnet mask |
| The phone's subnet mask.
Note: This parameter is disabled when DHCP is enabled. | |
| IP Gateway | Dotted-decimal IP address |
| The phone's default router. | |
| VLAN | |
| See VLAN Menu . | |
| 802.1X Authentication | Enabled, Disabled |
| If enabled, the phone will use the 802.1 Authentication parameters to satisfy the negotiation requirements for each EAP type. | |
| 802.1X Menu | |
| See 802.1X Menu . | |
| Storm Filtering | Enabled, Disabled |
| If enabled, received Ethernet packets are filtered so that the TCP/IP stack does not process bad data or too much data.
The default value is Enabled. | |
| LAN Port Mode | 0 = Auto, 1 = 10HD, 2 = 10FD, 3 = 100HD, 4 = 100FD, 5 = 1000FD |
| The network speed over Ethernet. The default value is Auto. HD means half duplex and FD means full duplex.
Note: Polycom recommends that you do not change this setting. | |
| PC Port Mode | 0 = Auto, 1 = 10HD, 2 = 10FD, 3 = 100HD, 4 = 100FD, 5 = 1000FD, -1 = Disabled |
| The network speed over Ethernet. The default value is Auto. HD means half duplex and FD means full duplex.
Note: Polycom recommends that you do not change this setting unless you want to disable the PC port. | |

| <i>Name</i> | <i>Possible Values</i> |
|--|--------------------------------|
| 1000BT LAN Clock | 0=Auto 1=Slave 2=Master |
| <p>The mode of the LAN clock.</p> <p>The default value is Slave (this device receives its clock timing from a master device).</p> <p>Note: Polycom recommends that you do not change this setting unless you have Ethernet connectivity issues. This setting was chosen to give the best results from an EMI perspective.</p> | |
| 1000BT PC Clock | 0=Auto 1=Slave 2=Master |
| <p>The mode of the PC clock. The default value is Auto.</p> <p>Note: Polycom recommends that you do not change this setting unless you have Ethernet connectivity issues. This setting was chosen to give the best results from an EMI perspective.</p> | |



Note: LAN Port Mode Support

The LAN Port Mode applies to all phones supported by SIP 3.0. The PC Port Mode parameters are available only on phones with a second Ethernet port. Only the SoundPoint IP 560 and VVX 1500 phones support the LAN Port Mode and PC Port Mode setting of 1000FD. The 1000BT LAN Clock and 1000BT PC Clock parameters are available only on SoundPoint IP 560 phones.

VLAN Menu

You can modify the following parameters in the VLAN menu:

Table 3-7: VLAN Menu

| <i>Name</i> | <i>Possible Values</i> |
|--|-----------------------------|
| VLAN ID | Null, 0 through 4094 |
| <p>The phone's 802.1Q VLAN identifier. The default value is Null.</p> <p>Note: Null = no VLAN tagging</p> | |
| VLAN Filtering | Enabled, Disabled |
| <p>If enabled, received Ethernet packets are filtered so that the TCP/IP stack does not process invalid data or too much data. The default value is Disabled.</p> <p>Note: VLAN filtering is not supported on the VVX family of phones.</p> <p>Note: VLAN filtering is enabled by default on the SpectraLink handsets.</p> | |

| <i>Name</i> | <i>Possible Values</i> |
|---|--|
| LLDP | Enabled, Disabled |
| <p>If enabled, the phone will use the LLDP protocol to communicate with the network switch for certain network parameters. Most often this will be used to set the VLAN that the phone should use for voice traffic. It also reports power management to the switch. The default value is Enabled.</p> <p>For more information on how to set VLAN and LLDP, see LLDP and Supported TLVs.</p> <p>Note: The SpectraLink handsets do not support LLDP.</p> | |
| CDP Compatibility | Enabled, Disabled |
| <p>If enabled, the phone will use CDP-compatible signaling to communicate with the network switch for certain network parameters. Most often this will be used to set the VLAN that the phone should use for Voice Traffic, and for the phone to communicate its PoE power requirements to the switch. The default value is Enabled.</p> <p>Note: The SpectraLink handsets do not use CDP.</p> | |
| VLAN Discovery | 0=Disabled, 1=Fixed (default), 2=Custom |
| <p>For a detailed description, see Assigning a VLAN ID Using DHCP.</p> <p>Disabled: No VLAN discovery through DHCP.</p> <p>Fixed: Use predefined DHCP vendor-specific option values of 128, 144, 157 and 191. If one of these is used, <i>VLAN ID Option</i> will be ignored</p> <p>Custom: Use the number specified for <i>VLAN ID Option</i> as the DHCP private option value.</p> <p>Note: The SpectraLink handsets do not use VLAN Discovery.</p> | |
| VLAN ID Option | 128 through 254 (Cannot be the same as Boot Server Option) (default is 129) |
| <p>The DHCP private option (when <i>VLAN Discovery</i> is set to Custom).</p> <p>For more information, see Assigning a VLAN ID Using DHCP.</p> | |

802.1X Menu

The 802.1X Menu only appears if 802.1X authentication is enabled. For more information, see [Provisioning SpectraLink 8400 Series Wireless Handsets](#).

The following 802.1X configuration parameters can be modified from the 802.1X menu:

Table 3-8: 802.1X Menu

| <i>Name</i> | <i>Possible Values</i> |
|---|---|
| EAP Method | 0 = None, 1=EAP-TLS, 2=EAP-PEAPv0/MSCHAPv2, 3=EAP-PEAPv0/GTC, 4=EAP-TTLS/EAP-MSCHAPv2, 5=EAP-TTLS/EAP-GTC, 6=EAP-FAST, 7=EAP-MD5 |
| <p>The selected EAP type to be used for authentication. For more information, see Supporting 802.1X Authentication.</p> | |

| <i>Name</i> | <i>Possible Values</i> |
|--|-----------------------------|
| Identity | UTF-8 encoded string |
| The identity (or user name) required for 802.1X authentication. | |
| Password | UTF-8 encoded string |
| The password required for 802.1X authentication. The minimum length is 6 characters. | |
| Anonymous ID | UTF-8 encoded string |
| The anonymous user name for constructing a secure tunnel for tunneled authentication and FAST authentication. | |
| PAC File Info | |
| See PAC File Information . | |
| EAP-FAST Inband Provisioning | Enabled, Disabled |
| A flag to determine whether EAP-FAST Inband Provisioning is enabled. This parameter is used only if <i>EAP Method</i> is EAP-FAST. | |

PAC File Information

You can modify Protected Access Credential (PAC) File Information from the PAC File Information menu:

Table 3-9: PAC File Information Menu

| <i>Name</i> | <i>Possible Values</i> | <i>Description</i> |
|-------------------|------------------------|---|
| PAC File Password | UTF-8 encoded string | The password required to decrypt the PAC file. |
| PAC File Name | UTF-8 encoded string | The path or URL of the PAC file for download. |
| Remove PAC File | UTF-8 encoded string | A flag to determine whether or not to delete the PAC file from the phone. |

Wi-Fi Menu

Currently the Wi-Fi menu displays only on the SpectraLink handsets.

You can modify the following parameters from the Wi-Fi menu:

Table 3-10: Wi-Fi Menu

| <i>Name</i> | <i>Possible Values</i> |
|---|---|
| Enabled | Yes, No |
| A flag to determine if the wireless interface is enabled or not. | |
| DHCP | Enabled, Disabled |
| If enabled, DHCP will be used to obtain the parameters discussed in DHCP or Manual TCP/IP Setup. | |
| DHCP Boot Server | Enabled, Disabled |
| A flag to determine if the DHCP server is accessible. | |
| IP Address | Dotted-decimal IP address |
| The phone's IP address. | |
| Note: This option is not available when the DHCP parameter is Enabled. | |
| Subnet Mask | Dotted-decimal subnet mask |
| The phone's subnet mask. | |
| Note: This option is not available when the DHCP parameter is Enabled. | |
| IP Gateway | Dotted-decimal IP address |
| The phone's default router. | |
| CCX AP Required | Yes, No |
| A flag to determine if phones will connect to APs (access points) that do not advertise Cisco® Compatible Extensions (CCX v4) or higher. | |
| AC Required | Yes, No |
| A flag to determine if phones will connect only to APs (access points) that enforce access control (Wi-Fi Multimedia Admission Control [WMM-AC]). | |
| SSID | string |
| The Service Set Identifier (SSID) of the wireless network. | |
| Security | 0=No security, 1=WEP, 2=WPA-PSK, 3=WPA2-PSK, 4=WPA2-Enterprise |
| The wireless security mode. | |
| WEP | |
| See WEP Menu . | |
| WPA(2)-PSK | |
| See WPA (2) PSK Menu . | |
| WPA2-Enterprise | |
| See WPA2-Enterprise Menu . | |

| <i>Name</i> | <i>Possible Values</i> |
|----------------------------------|------------------------|
| Radio | |
| See Radio Menu . | |

WEP Menu

You can modify the following Wired Equivalent Privacy (WEP) configuration parameters on the WEP menu:

Table 3-21: WEP Menu

| <i>Name</i> | <i>Possible Values</i> |
|---|--|
| Authentication | 0=Open System (default), 1=Shared Key |
| The WEP authentication method. | |
| Key Length | 0=40 bits (default), 1=104 bits |
| The authentication key length. | |
| Default Key | 1 to 4 |
| The default key. The default key is 1. | |
| Encryption | Enabled, Disabled |
| A flag to determine if wireless data is encrypted. | |
| Key1, Key2, Key3, Key4 | Hexadecimal value |
| The authentication keys. There are four possible keys. The key length is determined by the Key Length parameter. | |

WPA (2) PSK Menu

You can modify the following Wi-Fi Protected Access (WPA)/WPA2 Pre-Shared Key (PSK) configuration parameters on the WPA(2)-PSK menu:

Table 3-32: WPA (2) PSK Menu

| <i>Name</i> | <i>Possible Values</i> |
|--------------------------|--|
| PSK Type | 0=Passphrase (default), 1=Hexadecimal key |
| The pre-shared key type. | |

| <i>Name</i> | <i>Possible Values</i> |
|--|---------------------------------------|
| Passphrase | 8 to 63 character ASCII string |
| The authentication passphrase. | |
| Note: This parameter is unavailable when PSK Type is 1. | |
| Key | 256 bit hexadecimal string |
| The authentication key. | |
| Note: This parameter is unavailable when PSK Type is 0. | |

WPA2-Enterprise Menu

You can modify the following parameters from the WPA2-Enterprise menu:

Table 3-4: WPA2-Enterprise Menu

| <i>Name</i> | <i>Possible Values</i> |
|---|---|
| Fast Roaming Method | 0=Opportunistic Key Caching (OKC) , 1= Cisco Centralized Key Management (CCKM) |
| The fast roaming method. These fast roaming methods allow for the part of the key derived from the server to be cached in the wireless network, thereby, shortening the time to renegotiate a secure handoff. | |
| EAP Method | 2=EAP-PEAPv0/MSCHAPv2 (default), 6=EAP-FAST |
| The Extensible Authentication Protocol (EAP). | |
| User ID | String |
| The authentication user name. | |
| Password | String |
| The authentication password. | |
| PAC File Info | |
| See PAC File Information . | |
| EAP-FAST Inband Provisioning | Enabled, Disabled |
| A flag to determine whether or not EAP-FAST Inband Provisioning is enabled. Note: This parameter is unavailable when EAP Method is 2. | |

Radio Menu

You can modify the following parameters from the Radio menu:

Table 3-54: Radio Menu

| <i>Name</i> | <i>Possible Values</i> |
|---|--------------------------------|
| Regulatory Domain | 0, 1, 2, 4, 7, 8, or 10 |
| SpectralLink 8400 Series handsets only. Available values specify the regulatory domain. The supported values are 1 (North America), 2 (Europe), 4 (Singapore), 7 (Hong Kong), 8 (Mexico), and 10 (Australia). If Null, no regulatory domain is selected. You must set the regulatory domain before the handsets can be used. There is no default setting for this option and the handsets will not associate with an access point (AP) until you specify a value. | |
| 5 GHz | |
| See 5 GHz Menu . | |
| 2.4 GHz | |
| See 2.4 GHz Menu . | |

5 GHz Menu

You can modify the following parameters from the 5 GHz menu:

Table 3-65: 5 GHz Menu

| <i>Name</i> | <i>Possible Values</i> |
|---|--------------------------|
| 5 GHz Enable | Enabled, Disabled |
| A flag to determine if the 5 GHz band is enabled. | |
| Sub-bandx Enable | Enabled, Disabled |
| A flag to determine if the 5 GHz sub-band is enabled. There are four sub-bands (x=1 to 4). | |
| Sub-bandx Transmit Power | 1 to 7 |
| The maximum power that the handset uses to transmit in the 5 GHz sub-band. There are four sub-bands (x=1 to 4). For more information, see the <code>device.wifi.radio.band5GHz.subBandx.txPower</code> set of parameters in <device/> . | |

2.4 GHz Menu

You can modify the following parameters from the 2.4 GHz menu:

Table 3-7: 2.4 GHz Menu

| <i>Name</i> | <i>Possible Values</i> |
|---|--------------------------|
| 2.4 GHz Enable | Enabled, Disabled |
| A flag to determine if the 2.4 GHz band is enabled. | |
| 2.4 GHz Transmit Power | 1 to 7 |
| The maximum power that the handset uses to transmit in the 2.4 GHz band.
For more information, see <device/> . | |

USBNet Menu

Currently the USBNet menu displays only for the SpectraLink handsets.

You can modify the following parameters from the USBNet menu:

Table 3-87: USBNet Menu

| <i>Name</i> | <i>Possible Values</i> |
|---|-----------------------------------|
| Enabled | Yes, No |
| A flag to determine if USB networking is supported. | |
| IP Address | Dotted-decimal IP address |
| The handset's dotted-decimal IP address on the USBNet interface. For SpectraLink handsets, the default value is 169.254.1.2 . | |
| Subnet Mask | Dotted-decimal subnet mask |
| The phone's subnet mask. For SpectraLink handsets, the default value is 255.255.0.0 . | |
| IP Gateway | Dotted-decimal IP address |
| The phone's default router. For SpectraLink handsets, the default value is 169.254.1.1 . | |
| DHCP | Enabled, Disabled |
| If enabled, DHCP will be used to obtain the parameters discussed in DHCP or Manual TCP/IP Setup. | |

CMA Menu

The CMA Menu appears only if CMA provisioning is enabled. Currently, the CMA Menu only displays for the Polycom VVX 1500 phone. For more information, see [Provisioning VVX Phones Using a Polycom CMA System](#).

You can modify the following parameters from the CMA menu:

Table 3-9: CMA Menu

| <i>Name</i> | <i>Possible Values</i> |
|--|---|
| CMA Mode | Disabled, Static, Auto |
| Determines how the phone should retrieve the Polycom CMA server IP address. The possible values are: | |
| <ul style="list-style-type: none"> • Auto The phone must use SRV lookup to find the Polycom CMA server IP address. • Disabled The Polycom CMA server is not contacted. • Static The Polycom CMA server name or IP address is specified in device settings. | |
| Server Address | Dotted-decimal IP address OR Domain name string OR URL |
| The Polycom CMA server name or IP address. | |
| Login Credentials | |
| See Login Credentials Menu . | |

Login Credentials Menu

You can modify the following parameters from the Login Credentials menu:

Table 3-10: Login Credentials Menu

| <i>Name</i> | <i>Possible Values</i> |
|---|-----------------------------|
| Domain | UTF-8 encoded string |
| The domain name used by a server. | |
| User | UTF-8 encoded string |
| The user name used to authenticate to a server. | |
| Password | UTF-8 encoded string |
| The password used to authenticate to a server. | |

TLS Menu

This section refers to the TLS Menu available in the Updater, not UC Software. There is another menu, called TLS Security, available in the UC Software. You can modify the following parameters from the TLS Menu:

Table 3-11: TLS Menu

| <i>Name</i> | <i>Possible Values</i> |
|--|------------------------|
| Install Custom CA Cert | URL |
| A CA certificate that is installed on the phone to be used for TLS authentication. | |
| Install Custom Device Cert | URL |
| A device certificate installed on the phone to be used for Mutual TLS authentication. | |
| Clear Custom Device Cert | Yes, No |
| A flag to determine whether or not the device certificate can be removed from the phone. | |
| TLS Profile x | |
| There are currently two TLS Platform profiles. See TLS Profile Menu . | |
| Applications | |
| See Applications Menu . | |

TLS Profile Menu

You can modify the following parameters from the TLS Profile Menu:

Table 3-12: TLS Profile

| <i>Name</i> | <i>Possible Values</i> |
|---|------------------------|
| SSL Cipher Suite | String |
| The global cipher suite. | |
| Custom SSL Cipher Suite | String |
| A custom cipher suite. | |
| CA Cert List | String |
| The CA certificate sources that are valid for this profile. | |
| Device Cert List | String |
| The device certificate sources that are valid for this profile. | |

Applications Menu

You can modify the following parameters from the Applications Menu:

Table 3-13: Applications Menu

| <i>Name</i> | <i>Possible Values</i> |
|--|------------------------|
| 802.1X
The TLS Profile to use for 802.1X authentication. | 1 or 2 |
| Provisioning
The TLS Profile to use for provisioning authentication. | 1 or 2 |
| Syslog
The TLS Profile to use for syslog authentication. | 1 or 2 |

Syslog Menu

Syslog is a standard for forwarding log messages in an IP network. The term 'syslog' is often used for both the actual syslog protocol, as well as the application or library sending syslog messages.

The syslog protocol is a simple protocol: the syslog sender sends a small textual message (less than 1024 bytes) to the syslog receiver. The receiver is commonly called 'syslogd', 'syslog daemon' or 'syslog server'. Syslog messages can be sent through UDP, TCP, or TLS. The data is sent in cleartext.

Because syslog is supported by a wide variety of devices and receivers, syslog can be used to integrate log data from many different types of systems into a central repository.



Web Info: Information on Syslog

For more information on the syslog protocol, see [RFC 3164](#).

You can modify the following parameters from the Syslog Menu:

Table 3-143: Syslog Menu

| <i>Name</i> | <i>Possible Values</i> |
|--|--|
| Server Address | Dotted-decimal IP address OR Domain name string |
| The syslog server IP address. The default value is Null. | |
| Server Type | None=0, UDP=1, TCP=2, TLS=3 |
| The protocol that the phone will use to write to the syslog server. If set to None (or 0), transmission is turned off, but the server address is preserved. | |
| Facility | 0 to 23 |
| A description of what generated the log message. For more information, see section 4.1.1 of RFC 3164. The default value is 16, which maps to 'local 0'. | |
| Render Level | 0 to 6 |
| Specifies the lowest class of event that will be rendered to syslog. It is based on <code>log.render.level</code> and can be a lower value. See <log/> . | |
| Note: Use left and right arrow keys to change values. | |
| Prepend MAC Address | Enabled, Disabled |
| If enabled, the phone's MAC address is prepended to the log message sent to the syslog server. | |

Chapter 4: Setting Up the Provisioning Server

This chapter provides instructions for setting up your Polycom phones with a provisioning server. If you are new to this process, it is important to read every section in this chapter.

Because of the large number of optional installations and configurations that are available, this chapter focuses on one particular way that the Polycom® UC Software and the required external systems might initially be installed and configured in your network.

If you want to begin setting up phone features, go to [Part III: Configuring the Phone Features](#).

This chapter consists of the following sections:

- [Why Use a Provisioning Server](#)
- [Provisioning Server Security Notes](#)
- [Setting up an FTP Server as Your Provisioning Server](#)
- [Downloading Polycom UC Software Files to the Provisioning Server](#)
- [Deploying Polycom Phones](#)
- [Deploying and Updating Polycom Phones with a Provisioning Server](#)
- [Upgrading Polycom UC Software](#)
- [Supporting Legacy Phones](#)

This chapter also contains information on:

- [Provisioning VVX Phones Using a Polycom CMA System](#)
- [Provisioning SpectraLink 8400 Series Wireless Handsets](#)

Why Use a Provisioning Server?

Read this section if you have never set up a provisioning server before.

Polycom strongly recommends that you use a provisioning server to install and maintain your Polycom phones. You can set up a provisioning server on the local LAN or anywhere on the Internet. A provisioning server maximizes the flexibility you have when installing, configuring, upgrading, and maintaining the phones, and enables you to store configuration, log, directory, and override files on the server. If you allow the phone write access to your provisioning server, the phone can use the server to upload all of the file types and store administrator and user settings. The phone is designed such that if it cannot locate a provisioning server when it boots up, it will operate with internally saved parameters. This is useful when the provisioning server is not available.



Web Info: Registering Standalone Polycom Phones

If you want to register a single Polycom phone, see [Registering Standalone Polycom SoundPoint IP, SoundStation IP, and VVX 1500 Phones \(Quick Tip 44011\)](#).

You can configure multiple (redundant) provisioning servers—one logical server with multiple addresses—by mapping the provisioning server DNS name to multiple IP addresses. The default number of provisioning servers is one and the maximum number is eight. For more information on the protocol used, see [Supported Provisioning Protocols](#).

If you set up multiple provisioning servers, you must be able to reach all of the provisioning servers with the same protocol and the contents on each provisioning server must be identical. The parameters described in [Provisioning Server Menu](#) can be used to configure the number of times each server will be tried for a file transfer and also how long to wait between each attempt. You can configure the maximum number of servers to be tried. For more information, contact your Certified Polycom Reseller.

Provisioning Server Security Notes

Read this section if you have never set up a provisioning server before.

For organizational purposes, Polycom recommends configuring a separate log file directory, an override directory, a contact directory, and a license directory, though this is not required. Each directory can have different access permissions. For example, you can allow LOG, CONTACTS, and OVERRIDES to have full read and write access, and LICENSE to have read-only access.

You should ensure that the file permissions you create provide the minimum required access and that the account has no other rights on the server.



Tip: Allowing File Uploads to Your Provisioning Server

Polycom recommends that you allow file uploads to the provisioning server where the security environment permits. File uploads allow event log files to be uploaded. Log files provide backup copies of changes users make to the directory, and to the phone's configuration through the Web server and/or local user interface. These log files help Polycom provide customer support when diagnosing issues that may occur with the phone operation.

The phone's server account needs to be able to add files that it can write to in the log file directory and the provisioning directory. It must also be able to list files in all directories mentioned in the **<MAC-address>.cfg** file. All other files that the phone needs to read, such as the application executable and the standard configuration files, should be made read-only using file server file permissions.



Tip: Use RFC-Compliant Servers

Polycom recommends that you use RFC-compliant servers.

Each phone may open multiple connections to the server.

The phone will attempt to upload log files, a configuration override file, and a directory file to the server if changed. This requires that the phone's account has delete, write, and read permissions. The phone will still function without these permissions, but will not be able to upload files.

If you know the phone is going to download a file from the server, you should mark the file as read-only.

Setting up an FTP Server as Your Provisioning Server

Read this section if you have never set up a provisioning server before.

A simple provisioning configuration uses File Transfer Protocol or FTP. FTP servers are free, require installation, and use logins and passwords. A free and popular server, FileZilla Server, is available for Windows. FileZilla Server (version 0.9.xx) has been tested with the UC Software.



Tip: Choosing a Provisioning Protocol

By default, Polycom sets FTP as the provisioning protocol on all Polycom phones. This guide focuses on the FTP provisioning protocol. Other supported protocols include TFTP, HTTP, and HTTPS.

To set up an FTP server using FileZilla Server:

- 1 Download and install the latest version of [FileZilla Server](#).
- 2 After installation, a *Connect to Server* pop-up displays on your computer. Select **OK** to open the administrative user interface.
- 3 To configure a user, select **Edit > Users** in the status bar.
- 4 Select **Add**.
- 5 Enter the user name for the phone and select **OK**.
For example, *bill123*.
- 6 Select the **Password** checkbox and enter a password.
For example, *1234*. The phone will use this password to log in.
- 7 Select **Page > Shared folders** to specify the server-side directory where the provisioning files will be located (and the log files uploaded).
- 8 Select **Add** and pick the directory.
- 9 To allow the phone to upload logs onto the provisioning server, select the **Shared Folders > Files > select Write and Delete** checkboxes, and then select **OK**.
- 10 Determine the IP address of the FTP server by entering *cmd* in the Run dialog on your Start menu, and *ipconfig* in the command prompt.
The IP Address of the FTP server is shown.

Downloading Polycom UC Software Files to the Provisioning Server

This section explains how to download the Polycom Unified Communications (UC) Software to the provisioning server.

Go to the [Polycom UC Software Support Center](#) to download current and past releases, access supporting documentation; or, you navigate to the downloads and documents available for a specific product.

If you need to determine which Polycom UC Software release you need for your phones, refer to the [Polycom UC Software/Polycom SIP Software Release Matrix](#) to match your phone model to all UC Software releases and software components. You can download all UC Software from this Release Matrix.

Polycom provides the UC Software download in ZIP file format.

To download the Polycom UC Software:

- 1 Access Polycom UC Software from the [Polycom UC Software Support Center](#) or the [Polycom UC Software/Polycom SIP Software Release Matrix](#).

Choose the combined UC Software package or the split UC Software package. Both packages are provided in ZIP file format. The split software package is smaller and simplifies the provisioning process. Whereas the combined version contains all files for all phone models, the split version contains **sip.Id** files for each phone model, enabling you to choose files for your phone model and store software versions for different phone models in the same directory.

- 2 Acknowledge that you read the notices, accept the agreement, and choose **Submit**.
- 3 Save the UC Software ZIP file download.
- 4 Extract (uncompress) the ZIP file.

Copy all files from the distribution ZIP file to the home directory on the provisioning server, maintaining the same folder hierarchy. To simplify provisioning, Polycom recommends editing copies of each file as a best practice to ensure that you have unedited template files containing the default values.

The split image file contains individual **sip.Id** files for each phone model as well as all of the template configuration files included in the combined image file. To find the **sip.Id** file for your phones, see [Product, Model, and Part Number Mapping](#).

For a list and brief description of all available template files included with Polycom UC Software 4.1.0, see [Using the Template Configuration Files](#).



Note: See the Release Notes for a Description of all Parameters for a UC Software Release

For a description of each file in a UC Software distribution, see the *UC Software Release Notes* for a particular UC Software release on the [Polycom UC Software Support Center](#) or from the [Polycom UC Software/Polycom SIP Software Release Matrix](#).

Deploying and Updating Polycom Phones with a Provisioning Server

This section explains how to deploy and update Polycom phones from a provisioning server. If you are provisioning the phones using a provisioning server for the first time, follow the provisioning process described in the section [Deploying Polycom Phones Using a Provisioning Server](#). If you are provisioning a phone in one of the following special scenarios, refer to the relevant section:

- If your organization uses the Polycom® Converged Management Application™ (CMA™) system, read [Provisioning VVX Phones Using a Polycom CMA System](#) to understand the different provisioning options available for your organization's VVX 1500 phones.

- If you are provisioning SpectraLink handsets, read [Provisioning SpectraLink 8400 Series Wireless Handsets](#) to understand the different provisioning options available for your organization's handsets.

As of Polycom UC Software 4.0.1, the Updater and UC Software are packaged together for all phones except the SoundStation IP 6000.

As of Polycom UC Software 3.3.0, Polycom phones can boot up without any configuration files; however, you will need to configure certain parameters in the configuration files - for example, a registration address, label, and SIP server address - to make the phones usable.

You can create as many configuration files as you want and your configuration files can contain any combination of parameters you put in them. You can put all parameters into one file or, for example, you can put SIP server parameters in one file and phone features parameters in another file. For detailed information on how to use the configuration files, see [Using the Template Configuration Files](#) in *Chapter 5: Configuration Methods*.

For large-scale deployments, the centralized provisioning method using configuration files is strongly recommended. For smaller scale deployments, the Web Configuration Utility or local interface may be used, but administrators need to be aware that settings made using these methods will override settings made using configuration files.

For instructions on how to encrypt your configuration files, see [Encrypting Configuration Files](#).

Deploying Polycom Phones with a Provisioning Server

To deploy phones with a provisioning server:

- 1 Obtain a list of MAC addresses for the phones you want to deploy.

The MAC address is a 12-digit hexadecimal number on a label on the back of the phone and on the outside of the shipping box.

- 2 Create a per-phone **phone<MACAddress>.cfg** file.



Tip: Choosing the File Name for a Per-Phone Configuration File

Do not use the following file names as your per-phone file name: **<MACAddress>-phone.cfg**, **<MACAddress>-Web.cfg**, **<MACAddress>-app.log**, **<MACAddress>-boot.log**, or **<MACAddress>-license.cfg**. These file names are used by the phone itself to store user preferences (overrides) and logging information.

Add phone registration parameters to the file, for example `reg.1.address`, `reg.1.label`, and `reg.1.type`.

- 3 Create a per-site **site<location>.cfg** file.

For example, add the SIP server or feature parameters like `voIpProt.server.1.address` and `feature.corporateDirectory.enabled`.



Settings: Configuring Your Phone for Local Conditions

Most of the default settings are typically adequate; however, if SNTP settings are not available through DHCP, you will need to edit the SNTP GMT offset, and (possibly) the SNTP server address for the correct local conditions. Changing the default daylight savings parameters will likely be necessary outside of North America. Disable the local Web (HTTP) server or change its signaling port if the local security policy dictates (see [<httpd/>](#)). Change the default location settings for user interface language and time and date format (see [<lcl/>](#))

- 4 Create a master configuration file by performing the following steps:
 - a Enter the name of each per-phone and per-site configuration files created in steps 2 and 3 in the CONFIG_FILES attribute of the master configuration file (**000000000000.cfg**). For help using the master configuration file, see [Understanding the Master Configuration File](#) in Chapter 5.
For example, add a reference to **phone<MACaddress>.cfg** and **sip650.cfg**.
 - b (Optional) Edit the LOG_FILE_DIRECTORY attribute of master configuration file so that it points to the log file directory.
 - c (Optional) Edit the CONTACT_DIRECTORY attribute of master configuration file so that it points to the organization's contact directory.
 - d (Optional) Edit the USER_PROFILES_DIRECTORY attribute of master configuration file if you intend to enable the User Login feature.
For more information, see [Using User Profiles](#).
 - e (Optional) Edit the CALL_LISTS_DIRECTORY attribute of master configuration file so that it points to the user call lists.
- 5 Perform the following steps to configure the phone to point to the IP address of the provisioning server and set up the user:
 - a On the phone's Home screen or idle display, select **Settings > Advanced > Admin Settings > Network Configuration**.
When prompted for the administrative password, enter **456**. The Provisioning Server entry is highlighted.
 - b Press the **Select** soft key.
 - c Scroll down to **Server Type** and ensure that it is set to **FTP**.
 - d Scroll down to **Server Address** and enter the IP address of your provisioning server.
Press the **Edit** soft key to edit the value and the **OK** soft key to save your changes.
 - e Scroll down to Server User and Server Password and enter the user name and password of the user you created on your provisioning server.

In [Setting up an FTP Server as Your Provisioning Server](#) the example user given was *bill1234* and the example password was *1234*.

f Press the **Back** soft key twice.

g Scroll down to **Save & Reboot**, and then press the **Select** soft key.

The phone reboots.

The UC Software modifies the APPLICATION APP_FILE_PATH attribute of the master configuration file so that it references the appropriate **sip.Id** files. For example, the reference to **sip.Id** is changed to **2345-12600-001.sip.Id** to boot the SoundPoint IP 650 image.

After this step, the UC Software will try the unmodified APPLICATION APP_FILE_PATH attribute.

At this point, the phone sends a DHCP Discover packet to the DHCP server. This is found in the Bootstrap Protocol/option 'Vendor Class Identifier' section of the packet and includes the phone's part number and the BootROM version.

For example, a SoundPoint IP 650 might send the following information:

```
5EL@ DC?5cSc52*46*(9N7*<u6=pPolycomSoundPointIP-SPIP_6502345-12600-001,1BR/4.0.0.0155/23-May-07 13:35BR/4.0.0.0155/23-May-07 13:35
```

For more information, see [Parsing Vendor ID Information](#).

6 Ensure that the configuration process completed correctly.

On the phone, press the **Menu** key, and then select **Status > Platform > Application** to see the UC Software version and **Status > Platform > Configuration** to see the configuration files downloaded to the phone.

Monitor the provisioning server event log and the uploaded event log files (if permitted). All configuration files used by the provisioning server are logged.

The phone will upload two logs files to the LOG_DIRECTORY directory:

<MACaddress>app.log and **<MACaddress>boot.log**.

You can now instruct your users to start making calls.

Upgrading Polycom UC Software

You can upgrade the software that is running on the Polycom phones in your organization. The upgrade process varies with the version of Polycom UC Software that is currently running on your phones and with the version that you want to upgrade to. The Updater, UC Software executable, and configuration files can all be updated using centralized provisioning.



Admin Tip: Updating UC Software on a Single Phone

You can use the Software Upgrade tool in the Web Configuration Utility to update the UC Software version running on a single phone. Note that configuration changes made to individual phones using the Web Configuration Utility will override configuration settings made using central provisioning. For instructions on how to update UC Software, see [Using the Software Upgrade Tool in the Web Configuration Utility \(Feature Profile 67993\)](#).



Web Info: Downgrading from UC Software 4.0.0 or later

Once you have deployed the phones using UC Software 4.0.0 or later, you can downgrade to a previous software release by following the instructions in [Technical Bulletin 64731: Upgrading Polycom Phones to and Downgrading Phones From Polycom UC Software 4.0.0](#).

To continue setting up a provisioning server, most administrators can use the instructions shown in the next section, [Upgrading Current Phones to UC Software 4.1.0](#). If you provisioned a VVX 1500 phone using CMA, see [Upgrading Polycom UC Software Using Polycom CMA](#).

If you are using legacy phones—including SoundPoint IP 300, 301, 320, 330, 430, 500, 501, 600, 601, and 670, and/or SoundStation IP 4000 and 7000 phones—along with other models, you will need to change the phone configuration files to support these legacy phones when software releases UC Software 4.1.0 or later are deployed. To provision legacy phones, go to [Supporting Legacy Phones](#). The following models were discontinued:

- The SoundPoint IP 300 and 500 phones as of May 2006
- The SoundPoint IP 301, 600, and 601 phones as March 2008
- The SoundPoint IP 501 phone as of August 2009
- The SoundStation IP 4000 phone as of May 2009
- The SoundPoint IP 430 phone as of April 2010
- The SoundPoint IP 320 and 330 phones as of December 2009

The following models will no longer be supported by new releases of UC Software:

- The SoundStation IP 7000 phone
- The SoundPoint IP 670 phone

Upgrading Current Phones to UC Software 4.1.0

If your Polycom phones are running UC Software 4.0.x, you can upgrade to UC Software 4.1.0 by following the instructions.

If your phones are running a software release earlier than UC Software 4.0.0, you can upgrade to UC Software 4.0.1 by following the instructions in [Technical Bulletin 64731: Upgrading Polycom Phones to and Downgrading Phones From Polycom UC Software 4.0.0](#).

To update phones to Polycom UC Software 4.0.1:

- 1 Back up your existing application and configuration files.
- 2 Create your new configuration using UC Software 4.1.0.
Configuration file changes and enhancements are explained in the Release Notes that accompany the software.



Caution: Mandatory Changes to Configuration Files

To ensure predictable phone behavior, the configuration files listed in CONFIG_FILES attribute of the master configuration file must be updated when the software is updated. You will need to add new configuration files to the CONFIG_FILES attribute in the appropriate order.

- 3 Save the new configuration files and images (such as **sip.ld**) on your provisioning server.
- 4 Reboot the phones using an automatic method such as polling or check-sync.
The phones can be rebooted remotely through the SIP signaling protocol.
See [<volP.SIP.specialEvent.*/>](#).

The phones can be configured to periodically poll the provisioning server for changed configuration files or application executables. If a change is detected, the phone may reboot to download the change.



Tip: Rebooting Your Phone

You should only reboot your phone using the multiple-key combination as a backup option if another reboot method fails. For details on using a multiple key combination to reboot your phone, see [Multiple Key Combinations](#).

You can reboot phones remotely through the SIP signaling protocol. See the parameter [volpProt.SIP.specialEvent.checkSync.alwaysReboot](#) in [<volpProt/>](#).

You can configure the phones to periodically poll the provisioning server to check for changed configuration files or application executable. If a change is detected, the phone will reboot to download the change. See [prov.polling.*](#).

Supporting Legacy Phones

With enhancements available since BootROM 4.0.0 and SIP 2.1.2, you can specify a different software and configuration files for legacy phones in the same master configuration file you are using for your non-legacy phones.

Polycom UC Software 4.1.0 or later software distributions contain only the files for that release. To use a software version for a legacy phone in the same master configuration file, you need to specify the phone product name, phone model number, or phone part number, and rename the legacy phone configuration files.

To get the phone product name, phone model number, or phone part number for a phone, see [Product, Model, and Part Number Mapping](#).

You must rename the *sip.ld*, *sip.cfg*, and *phone1.cfg* from a previous 2.1.x distribution that is compatible with SoundPoint IP 300 and 500 phones, or a previous 3.1.y distribution that is compatible with SoundPoint IP 301, 501, 600, 601, and SoundStation IP 4000 phones, or a previous 3.2.z distribution that is compatible with SoundPoint IP 430 phones, or a previous 3.3.w distribution that is compatible with SoundPoint IP 320 and 330 phones, or a previous 4.0.v distribution that is compatible with SoundPoint IP 670 and SoundStation IP 7000 phones.

Supported Software for Legacy Phones

The SoundPoint IP 300 and 500 phones will be supported on the latest maintenance patch release of the SIP 2.1 software stream—currently SIP 2.1.4. Any critical issues that affect SoundPoint IP 300 and 500 phones will be addressed by a maintenance patch on this stream until the End of Life date for these products. Phones should be upgraded to BootROM 4.0.0 for these changes to be effective.

The SoundPoint IP 301, 501, 600, and 601, and the SoundStation IP 4000 phones will be supported on the latest maintenance patch release of the SIP 3.1 software stream—currently SIP 3.1.7. Any critical issues that affect SoundPoint IP 300 and 500 phones will be addressed by a maintenance patch on this stream until the End of Life date for these products. Phones should be upgraded to BootROM 4.0.0 or later for these changes to be effective.

The SoundPoint IP 430 phone will be supported on the latest maintenance patch release of the SIP 3.2 software stream—currently SIP 3.2.4RevB. Any critical issues that affect SoundPoint IP 430 phones will be addressed by a maintenance patch on this stream until the End of Life date for these products. Phones should be upgraded to BootROM 4.2.2 for these changes to be effective.

The SoundPoint IP 320 and 330 phones will be supported on the latest maintenance patch release of the UC Software 3.3.1 software stream—currently UC Software 3.3.1RevF. Any critical issues that affect SoundPoint IP 320 and 330 phones will be addressed by a maintenance patch on this stream until the End of Life date for these products. Phones should be upgraded to BootROM 4.0.0 for these changes to be effective.

The SoundPoint IP 670 and SoundStation IP 7000 are supported by UC Software 4.0.x - currently 4.0.2B - and are not supported by UC Software 4.1.x or later. Critical issues that affect these phones will be addressed by a maintenance patch on the 4.0.x stream until the End of Life date for these products. Phones should be upgraded to BootROM 4.0.0 for these changes to be effective.

If you are upgrading legacy phones to UC Software 4.0.x, Polycom recommends modifying your configuration files in the following way.

To upgrade legacy phones to UCS 4.1.x:

- 1 Do one of the following steps:
 - a Place all **bootrom.ld** files corresponding to the BootROM release zip file onto the provisioning server.
 - b Ensure that all phones are running BootROM 4.x.x or later.
- 2 Copy **sip.ld** (or the appropriate individual **sip.ld** from the split image file) from the UC Software 4.1.0 or later release distribution onto the provisioning server.
- 3 Rename the **sip.ld**, **sip.cfg**, and **phone1.cfg** from the previous distribution to one of the following sets of names, depending on the phone models that you are provisioning:
 - For SoundPoint IP 300 and 500 phones, rename the files to **sip_21x.ld**, **sip_21x.cfg**, and **phone1_21x.cfg** respectively.
 - For SoundPoint IP 301, 501, 600, 601, and SoundStation IP 4000 phones, rename the files to **sip_31y.ld**, **sip_31y.cfg**, and **phone1_31y.cfg** respectively.
 - For SoundPoint IP 430 phones, rename the files to **sip_323.ld**, **sip_323.cfg**, and **phone1_323.cfg** respectively.
 - For SoundPoint IP 320 and 330 phones, rename **sip.ld** to **sip_33x.ld** and add *_33x to the end of each **.cfg** configuration files (for example, rename **phone1.cfg** to **phone1_33x.cfg** and rename **sip-basic.cfg** to **sip-basic_33x.cfg**).
- 4 Save the renamed configuration files to the provisioning server.
- 5 Modify the **000000000000.cfg** file, if required, to match your configuration file structure.

The following illustration shows an example file:

| | |
|-------------------------|-----------------------------|
| CONTACTS_DIRECTORY | |
| LICENSE_DIRECTORY | |
| USER_PROFILES_DIRECTORY | |
| CALL_LISTS_DIRECTORY | |
| APPLICATION_SPIP300 | |
| APP_FILE_PATH_SPIP300 | sip_213.ld |
| CONFIG_FILES_SPIP300 | phone1_213.cfg, sip_213.cfg |
| APPLICATION_SPIP500 | |
| APP_FILE_PATH_SPIP500 | sip_213.ld |
| CONFIG_FILES_SPIP500 | phone1_213.cfg, sip_213.cfg |
| APPLICATION_SPIP301 | |
| APP_FILE_PATH_SPIP301 | sip_318.ld |
| CONFIG_FILES_SPIP301 | phone1_318.cfg, sip_318.cfg |
| APPLICATION_SPIP320 | |
| APP_FILE_PATH_SPIP320 | sip_334.ld |
| CONFIG_FILES_SPIP320 | |
| APPLICATION_SPIP330 | |
| APP_FILE_PATH_SPIP330 | sip_334.ld |
| CONFIG_FILES_SPIP330 | |
| APPLICATION_SPIP430 | |
| APP_FILE_PATH_SPIP430 | sip_327.ld |
| CONFIG_FILES_SPIP430 | phone1_327.cfg, sip_327.cfg |
| APPLICATION_SPIP501 | |
| APP_FILE_PATH_SPIP501 | sip_318.ld |
| CONFIG_FILES_SPIP501 | phone1_318.cfg, sip_318.cfg |
| APPLICATION_SPIP600 | |
| APP_FILE_PATH_SPIP600 | sip_318.ld |
| CONFIG_FILES_SPIP600 | phone1_318.cfg, sip_318.cfg |
| APPLICATION_SPIP601 | |
| APP_FILE_PATH_SPIP601 | sip_318.ld |
| CONFIG_FILES_SPIP601 | phone1_318.cfg, sip_318.cfg |
| APPLICATION_SPIP670 | |
| APP_FILE_PATH_SPIP670 | sip_402.ld |
| CONFIG_FILES_SPIP670 | |
| APPLICATION_SSIP4000 | |
| APP_FILE_PATH_SSIP4000 | sip_318.ld |
| CONFIG_FILES_SSIP4000 | phone1_318.cfg, sip_318.cfg |
| APPLICATION_SSIP7000 | |
| APP_FILE_PATH_SSIP7000 | sip_402.ld |
| CONFIG_FILES_SSIP7000 | |

- 6 Remove the **<MACAddress>.cfg** files on your provisioning server if the files correspond to legacy phones.

Using Variable Substitutions for Legacy Phones

You can support legacy phones using an enhancement that was added in BootROM 3.2.1/SIP 2.0.1. This enhancement enables you to use a phone-specific variable substitution [PHONE_MAC_ADDRESS] to name your configuration files and avoid the need to create a unique **<MACaddress>.cfg** file for each phone.

If you do not use a variable substitution, you will need to change all of the **<MACaddress>.cfg** files for SoundPoint IP 300, 301, 320, 330, 430, 500, 501, 600, and 601, and SoundStation IP 4000 phones. You will also need to make changes to all of the **<MACaddress>.cfg** files if you do not know which phones are SoundPoint IP 300, 301, 320, 330, 430, 500, 501, 600, and 601 or SoundStation IP 4000 models.

For details on using variable substitution as a configuration method, see Provisioning with the Master Configuration File (Best Practices 75907) on the [Polycom UC Software Support Center](#).

For more information, see [Technical Bulletin 35311: Maintaining Older Polycom Phones Beyond Their Last Supported Software Release](#).

Provisioning VVX 1500 Phones Using a Polycom CMA System

You can provision your organization's VVX 1500 phones and update the software using a Polycom CMA system. See the latest *UC Software Release Notes* on the [Latest Polycom UC Software Release](#) for Polycom UC Software and Polycom CMA for specific compatibility requirements and recommendations.

You can also provision your organization's VVX 1500 phones in a hybrid way, using both Polycom CMA and a provisioning server. In hybrid scenarios, settings made using the Polycom CMA have a higher priority than settings made using centralized provisioning. When the phone reboots, it will check the Polycom CMA system first for new software, and then check the provisioning server for configuration files and directories. Note that the phone will not check the provisioning server for software when CMA provisioning is enabled. To disable the CMA system, see [Disabling the Polycom CMA System](#).

In dynamic management mode, the Polycom CMA system can do the following:

- Configure your VVX 1500 phones using an automatic provisioning service
- Register your VVX 1500 phones with a standard-based presence service, so that presence states are shared with Polycom CMA contacts
- Provide your VVX 1500 phones with automatic software updates



Web Info: Provisioning VVX 1500 Phones using a Polycom CMA System

For more information about provisioning by a Polycom CMA system, see the [Polycom CMA System Operations Guide](#).

This section contains information on:

- [Provisioning Using Polycom CMA](#)
- [Disabling the Polycom CMA System](#)
- [Upgrading Polycom UC Software Using Polycom CMA](#)
- [Monitoring by Polycom CMA](#)

Provisioning Using Polycom CMA

In order to provision using the Polycom CMA system, the VVX phones must be installed with Polycom UC Software 3.3.1 or later.

Polycom CMA requires that the management application be installed on the same network to which your VVX 1500 phones are connected.

To configure the provisioning service settings on VVX 1500 phones:

- 1 Press the Menu key, and then select **Settings > Advanced > Administration Settings > Network Configuration > CMA Menu**.

You must enter the administrator password to access the network configuration. The factory default password is **456**.

- 2 Enter the following values:
 - **CMA Mode:** Select **Static** or **Auto**. If Auto: CMA picks the DNS name (ignores the Server Address – listed next). If Static, CMA uses the server address, listed next.
 - **Server Address:** Enter the address of the Polycom CMA system running the provisioning service. The address can be an IP address or a fully qualified domain name. For example, *123.45.67.890*.
- 3 Scroll to Login Credentials and tap the **Select** soft key. Enter the following values:
 - **CMA Domain:** Enter the domain for registering to the provisioning service. For example, *NorthAmerica*.



Tip: Domain When You Are Not Using Single Sign On

If you are not using a Single Sign On login with Active Directory on the Polycom CMA system, the phone will use the local accounts created on the Polycom CMA server.

- **CMA User:** Enter the user name for registering to the provisioning service. For example, *bsmith*.
 - **CMA Password:** Enter the password that registers the VX 1500 phone to the provisioning service (associated with the CMA user account). For example, *123456*.
- 4 Tap the **Back** soft key three times.
 - 5 Select **Save Config**.
The VVX 1500 phone reboots.



Tip: Configuring the Line Key with a Polycom CMA System

Only one phone line associated with a Polycom CMA system can be provisioned on a VVX 1500 phone, but the line key associated with that line is configurable. For more information on configuration file settings, see [<prov/>](#).



Web Info: Searching the CMA Directory

The user can now search for CMA users and groups in the CMA directory, place calls to those contacts, and view their presence status. For more information, see the [User Guide for the Polycom VVX 1500 Phone](#).

Disabling the Polycom CMA System

If you provision phones using the CMA system, and then want to disable that provisioning, use the following procedure.

To disable Polycom CMA provisioning:

- 1 Press the **Menu** key, and then select **Settings > Advanced > Administration Settings > Network Configuration > CMA Menu**.
You must enter the administrator password to access the network configuration. The default password is **456**.
- 2 In **CMA Mode:** select **Disable**.
- 3 Tap the **Back** soft key twice.
- 4 Select **Save Config**.
The VVX 1500 phone reboots.

Upgrading Polycom UC Software Using Polycom CMA

Software upgrades of the VVX 1500 phones are triggered by the Polycom CMA system as either automatic or scheduled updates.

Software update timer changes will not take effect until the next interval—after the current interval expires. For example:

- The current software update timer is set to 60 minutes.
- The provisioning by the Polycom CMA system fails.
- The software update timer is reset to five minutes (default). The five-minute timer is not set off until the last 60 minutes timer expires.

Monitoring by Polycom CMA

The following information is sent by the VVX 1500 phone to the Polycom CMA system:

- **Network adapter probe** This is the first message that the VVX 1500 phone sends to the Polycom CMA system. It provides the phone's IP address.
- **Software update check** This message provides the phone model, MAC address, and UC Software version currently running on the phone.
- **Software update status** This message provides confirmation of the phone's software upgrade.
- **Provisioning profile** This message requests configuration data for the phone so that the user can access the CMA directory, add CMA contacts to their Buddy list, and place audio and video calls to those contacts.
- **Provisioning status** This message provides confirmation of the receipt of the configuration data from the Polycom CMA system.
- **Call statistics** These messages are sent for all calls placed or answered by the phone's user.
- **Call end** This message is sent after all calls have ended.
- **Heartbeat data** This message contains status information about the phone. This message is sent to the Polycom CMA system periodically. How often the message is sent is configured by the administrator of the Polycom CMA system.
- **Events** This message provides information like gatekeeper registration events, presence registration events, and LDAP events to the Polycom CMA system.

Provisioning SpectraLink 8400 Series Wireless Handsets

Provisioning your organization's SpectraLink handsets in an 802.11 wireless environment requires you to follow several steps in addition to those required to provision the Polycom phones. These additional steps include:

- Setting up Access Points (APs) and Controllers
- Setting an Authentication Server

- Setting up a Wireless Configuration Station (WCS)



Web Info: Provisioning and Managing SpectraLink Handsets

For detailed information up provisioning and managing SpectraLink handsets, see the [Polycom SpectraLink 8400 Series Wireless Telephone Deployment Guide](#).

Chapter 5: Configuration Methods

This chapter explains three configuration methods you can use to configure settings and features on the phones:

- Centralized provisioning method (*for multiple phones*)
- Web Configuration Utility (*for a single phone*)
- Local phone user interface (*for a single phone*)

If you are using the VVX 1500, see [Provisioning VVX Phones Using a Polycom CMA System](#) for further provisioning information.

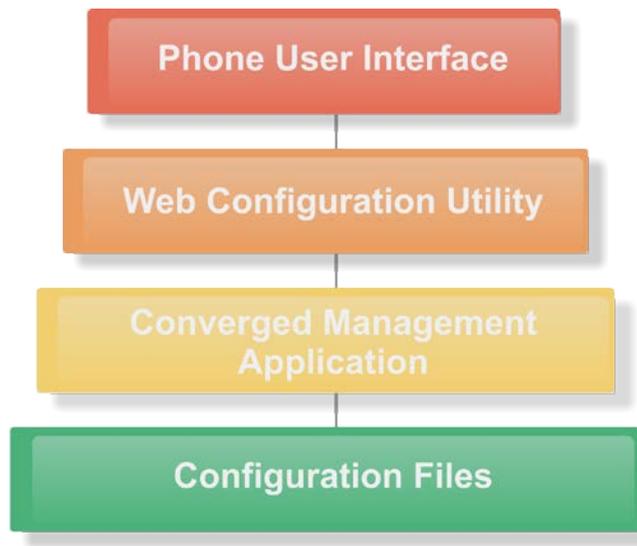
The three configuration methods explained in this chapter configure the phone features and settings detailed in [Part III: Configuring the Phone Features](#). Note that not all of the features and settings are available using each configuration method.



Web Info: Registering a Single Polycom Phone

If you want to register a single Polycom phone, see [Quick Tip 44011: Registering Standalone Polycom SoundPoint IP, SoundStation IP, and VVX 1500 Phones](#).

You can use one or more of these configuration methods in combination. If you use or plan to use multiple configuration methods, you will need to be aware of a hierarchy among the configuration methods - settings you make using a higher-priority method will override settings you make using a lower-priority method. See [Figure 5-1: Precedence Order Among the Configuration Methods](#) for an illustration of the precedence order among the provisioning methods.

Figure 5-1: Precedence Order Among Configuration Methods

Override Files

When you make a change to a phone's configuration using a higher-priority configuration method, an override file is created. For example, if you change settings using the Web Configuration Utility or the from the phone's user interface, the change in settings creates an override file and those settings are applied to the phone, overriding settings you made using the configuration files.



Troubleshooting: Understanding Configuration Priority Settings

If you or a user makes a configuration change using a higher-priority method, changes made to the same settings using a lower-priority method will not apply.

You can store the override files to the phone or, if you allow the phone write access to the provisioning server, have the override *file* uploaded to the provisioning server. If you permit the phone to upload to the provisioning server, the override settings will be saved as files named **<MAC Address>-phone.cfg** or **<MAC Address>-Web.cfg** and contain all of the changes that you or phone users make from their phone or using the Web Configuration Utility respectively. The advantage of allowing the phone write access to the provisioning server for override files is that user settings for a phone will survive restarts, reboots, and software upgrades. Note that if you allow the phone to write to the provisioning server, the phone uploads an override file the next time a configuration change is made from the phone. If you reformat the phone's file system, the override file will be deleted.

There are several ways to reset or clear features and settings being applied by an override file.

- On the phone, go to **Menu > Settings > Advanced > Administration Settings > Reset to Defaults**.

The following options display:

- **Reset Local Configuration** Clears the override file generated by changes using the phone user interface
- **Reset Web Configuration** Clears the override file generated by changes using the Web Configuration Utility.
- **Reset Device Settings** Resets the phone's flash file system settings that are not stored in an override file.
- **Format File System** Formats the phone's flash file system settings and deletes the UC Software application, log files, and override files. Note that if the override file is stored on the provisioning server, the phone will re-upload the override file when you provision the phone again. Formatting the phone's file system does not delete the device settings.
- **Reset to Factory** Formats the phone's flash file system and deletes the device settings.

The rest of this chapter explains each of the following configuration methods:

- [Using the Centralized Provisioning Method - Configuration Files](#)
- [Provisioning with the Web Configuration Utility](#)
- [Phone User Interface - Menu System Settings](#)

Using the Centralized Provisioning Method - Configuration Files

Polycom recommends using a central provisioning server when your VoIP environment has multiple phones. Polycom provides template configuration files in XML format that you can use to create a set of phone features and settings specific to your organization. All of the phone features and settings are outlined in [Part III: Configuring the Phone Features](#) of this Administrators' Guide. The UC Software configuration files you use to configure the phones are very flexible. Parameters can be stored in the files in any order and can be placed in any number of files. You can change the XML tree structure, move parameters around within the XML files, change the file names, or create your own configuration files. These files dictate the behavior of the phone once it is running the Polycom UC Software. Be aware that the configuration files have default values that you may want to change.



Settings: Using the Default Value for a Configuration Parameter

The phone will use the default value for a configuration parameter as long as the parameter has not been configured from any other source. Parameters can be changed using the local phone user interface, the Web configuration utility, a Polycom CMA system, and configuration files hosted on a central provisioning server.

Applying configuration files to phones from a central provisioning server enables you to apply a single set of parameters and settings to all of the phones in your deployment. The configuration files maximize flexibility in installing the UC Software, configuring the phones, and in upgrading and maintaining the phone settings over time.

Polycom phones can boot up without any configuration files; however, certain parameters will need to be changed for your phones to be usable within your organization. Note that if a phone cannot locate a provisioning server upon boot up, it will operate with internally stored default settings. To send and receive calls, you must specify a SIP server address and a registration address (the equivalent of a phone number) in the configuration files.

Note that as of Polycom UC Software 4.0.0, you can create user-specific configuration files that enable phone users to use their features and settings from any phone including those outside of your organization. To create a user-specific file, create a **<user>.cfg** on the provisioning server for the user (including default user accounts). For more information, see [Using User Profiles](#).

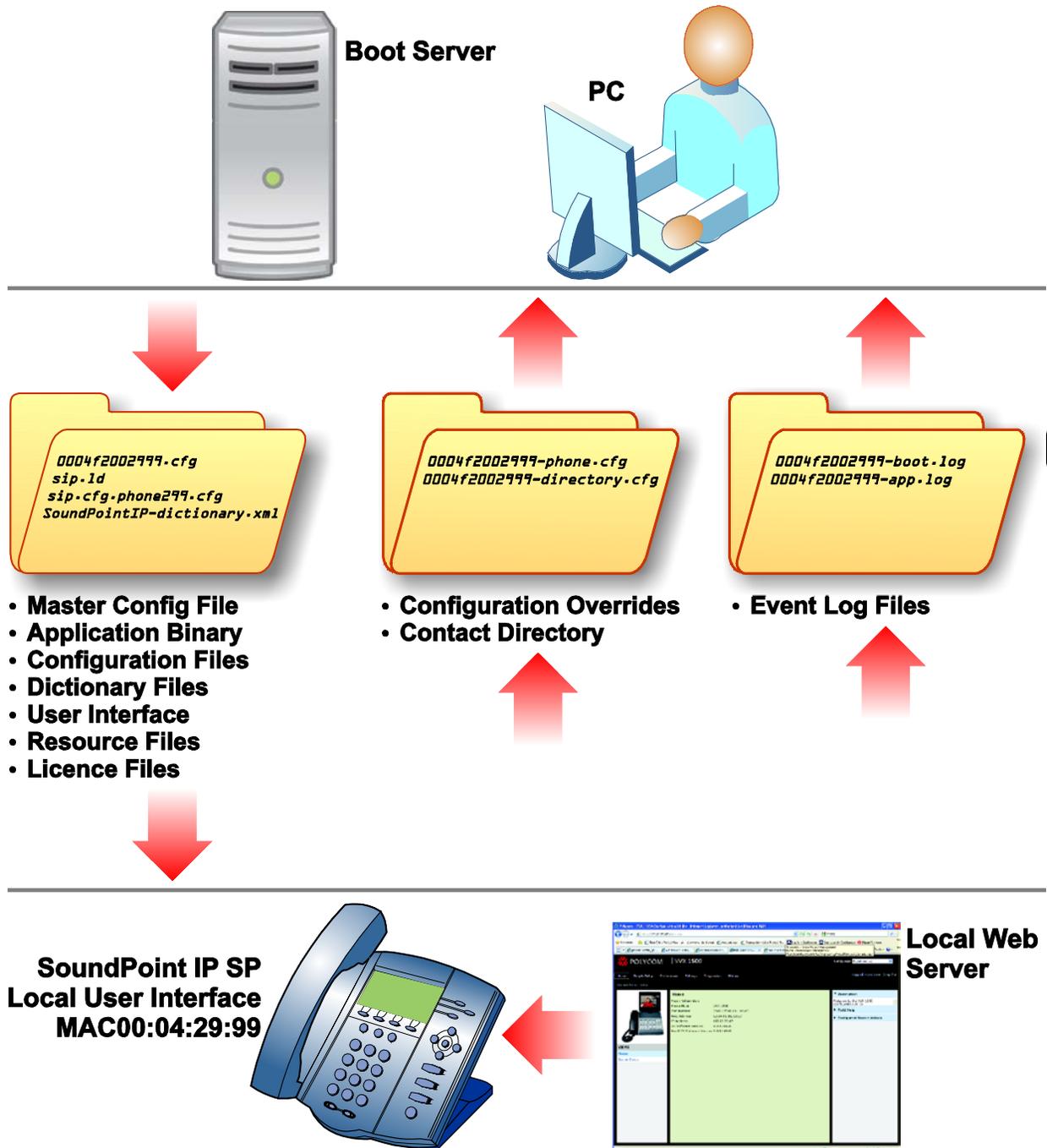


Settings: Choosing a Per-Phone Configuration File Name

Do not use **<MACaddress>-phone.cfg**, **<MACaddress>-Web.cfg**, **<MACaddress>-app.log**, **<MACaddress>-boot.log**, or **<MACaddress>-license.cfg** as the per-phone filename – where the MACaddress is represented as a 12-digit number (for example, 000123456789). These filenames are used by the phone itself to store user preference overrides and logging information.

The following figure shows an example of a phone network using the central provisioning method.

Figure 5-2: Network Layout Using Central Provisioning



Understanding the Master Configuration File

The centralized provisioning method requires you to use a master configuration file, named `000000000000.cfg` in the UC Software download. You can use the default master configuration

file or you can create and rename a master configuration file to apply to phones in a network in one of the following ways:

- To all of the phones in a deployment
- To a group of phones in a deployment
- On a per-phone basis (to a single phone)
- In a specific location



Settings: Use the `.cfg` extension on the master configuration file.

The master configuration file must have the `.cfg` extension. No other configuration files must have the `.cfg` extension.

Each of these ways is described next in more detail.

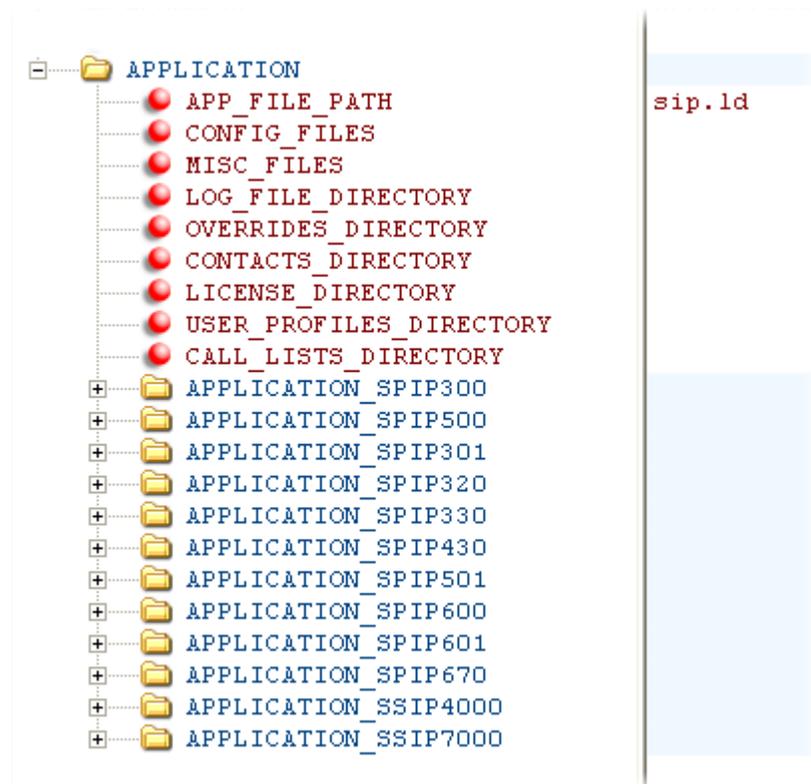
- **Default master configuration file** For deployments in which the configuration is identical for all phones, you can use the default master configuration file, named `000000000000.cfg` in the UC Software download, to configure all the phones in a deployment. Note that the phones are programmed to look first for their own `<MACAddress>.cfg` file and if a phone does not find a matching file, it looks next for the default file. If you do create and use a per-phone master configuration file, make a copy of the default file and rename it.
- **Group and per-phone master configuration file** If you want to apply features or settings to a group of phones within your deployment or to a single phone, make a copy of the default file and rename it. For a phone group, rename the file in a way that specifies the group-specific features or settings. For single phones, rename the file based on the phone's MAC address `<MACAddress>.cfg`. The MAC address, also known as the serial number (SN), is a unique a-f hexadecimal digit assigned to each phone. Note that you can use only lower-case digits, for example, `0004f200106c.cfg`. You can find the MAC address of a phone on a label on the back of the phone (under the Battery Pack on the SpectraLink handsets) or on the phone's menu system at **Menu > Status > Platform > Phone > S/N:** .
- **Specified master configuration file** You can specify a master configuration file in the provisioning server address, for example, `http://usr:pwd@server/dir/example1.cfg`. The filename must end with `.cfg` and be at least five characters long. If this file cannot be downloaded, the phone will search for a per-phone master configuration file, described next.



Settings: Pay Attention to Per-Phone File Names

Do not use the following names as extensions for per-phone files: **<MACaddress>-phone.cfg**, **<MACaddress>-Web.cfg**, **<MACaddress>-app.log**, **<MACaddress>-boot.log**, or **<MACaddress>-license.cfg**. These filenames are used by the phone to store override files and logging information.

The default master configuration file, **000000000000.cfg**, for Polycom UC Software 4.1.0 is shown next.



The following describes each of the master configuration file XML attributes and the APPLICATION directories.

- **APP_FILE_PATH** The path name of the UC Software application executable. The default value is `sip.ld`. Note that the phone automatically searches for the `sip.ld` and `<part number>.sip.ld`. This field can have a maximum length of 255 characters. If you want the phone to search for a `sip.ld` file in a location other than the default or use a different file name, or both, you can modify the default. For example, you can specify a URL with its own protocol, user name and password such as `http://usr:pwd@server/dir/sip.ld`.

- **CONFIG_FILES** Enter the names of your configuration files here as a comma-separated list. Each file name has a maximum length of 255 characters and the entire list of file names has a maximum length of 2047 characters, including commas and white space. If you want to use a configuration file in a different location or use a different file name, or both, you can specify a URL with its own protocol, user name and password, for example, *ftp://usr:pwd@server/dir/phone2034.cfg*.



Settings: Order of the Configuration Files

The order of the configuration files listed in CONFIG_FILES is significant:

- The files are processed in the order listed (left to right).
 - If the same parameter is included in more than one file or more than once in the same file, the first (left) parameter read is used.
-
- **MISC_FILES** A comma-separated list of other required files.
 - **LOG_FILE_DIRECTORY** An alternative directory to use for log files if required. A URL can also be specified. This is blank by default.
 - **CONTACTS_DIRECTORY** An alternative directory to use for user directory files if required. A URL can also be specified. This is blank by default.
 - **OVERRIDES_DIRECTORY** An alternative directory to use for configuration overrides files if required. A URL can also be specified. This is blank by default.
 - **LICENSE_DIRECTORY** An alternative directory to use for license files if required. A URL can also be specified. This is blank by default.
 - **USER_PROFILES_DIRECTORY** An alternative directory for the <user>.cfg files.
 - **CALL_LISTS_DIRECTORY** An alternative directory to use for user call lists if required. A URL can also be specified. This is blank by default.

The directories labeled **APPLICATION_SPIPXXX** indicate phone models that are not compatible with the latest UC Software version. If you are using any of the phone models listed in these directories, open the directory for the phone model you are deploying, and use the available fields to provision and configure those phones.

Understanding Variable Substitution

The master configuration template file, included in the UC Software files you download from the Polycom Voice Support Web site, is particularly important to the central provisioning method, which Polycom recommends using for large-scale deployments. There are two methods you can use to provision or configure phones with the master configuration file. The method you use depends on your deployment scenario. Understanding both methods enables you to deploy and manage your phones efficiently. For a detailed explanation of the two methods and their

advantages, see *Provisioning with the Master Configuration File* on the [Polycom UC Software Support Center](#).

You can also use variable substitution if you need to use different application loads on different phones on the same provisioning server by creating a variable in the master configuration file that is replaced by the MAC address of each phone when it reboots. You can use any of the following substitution strings:

- PHONE_MODEL
- PHONE_PART_NUMBER
- PHONE_MAC_ADDRESS

To find out the model number or part number of a product, see the section [Product, Model, and Part Number Mapping](#).

The following two examples illustrate the use of a variable substitution.



Web Info: Using the Master Configuration File

Using a variable substitution may simplify your overall provisioning. For a more detailed discussion of using a variable substitution in the master configuration file, see [Provisioning with the Master Configuration File \(Best Practices 75907\)](#) on the Polycom UC Software Support Center.

Example One

You can create a variable in the master configuration file that is replaced by the MAC address of each phone when it reboots.

The screenshot shows a file explorer on the left with a tree view under 'APPLICATION'. The tree includes several directories and files, each with a red circle icon: APP_FILE_PATH, CONFIG_FILES, MISC_FILES, LOG_FILE_DIRECTORY, OVERRIDES_DIRECTORY, CONTACTS_DIRECTORY, LICENSE_DIRECTORY, USER_PROFILES_DIRECTORY, CALL_LISTS_DIRECTORY, APPLICATION_SPIP300, APPLICATION_SPIP500, APPLICATION_SPIP301, APPLICATION_SPIP320, and APPLICATION_SPIP330. To the right, a code editor displays two lines of configuration code: `sip[PHONE_MAC_ADDRESS].ld` and `reg-basic[PHONE_MAC_ADDRESS].cfg`.

Example Two

You can direct phone update to a UC software build and configuration files based on the phone model number and part number. All XML attributes can be modified in this manner.

The screenshot shows a file explorer view of the 'APPLICATION' directory. The directory structure includes:

- APPLICATION
 - APP_FILE_PATH
 - CONFIG_FILES
 - MISC_FILES
 - LOG_FILE_DIRECTORY
 - OVERRIDES_DIRECTORY
 - CONTACTS_DIRECTORY
 - LICENSE_DIRECTORY
 - USER_PROFILES_DIRECTORY
 - CALL_LISTS_DIRECTORY
 - APPLICATION_SPIP300
 - APP_FILE_PATH_SPIP300
 - CONFIG_FILES_SPIP300
 - APPLICATION_SPIP500 (highlighted)
 - APP_FILE_PATH_SPIP500
 - CONFIG_FILES_SPIP500
 - APPLICATION_SPIP301
 - APPLICATION_SPIP320

To the right of the file explorer, the configuration files for each sub-directory are listed:

- For 'APPLICATION': sip.ld, reg-basic.cfg, sip-basic.cfg
- For 'APPLICATION_SPIP300': sip_213.ld, phone1_213.cfg, sip_213.cfg
- For 'APPLICATION_SPIP500': sip_213.ld, phone1_213.cfg, sip_213.cfg

Using the Template Configuration Files

You will find a number of template configuration files in the Polycom UC Software 4.0.1 download. Most configuration parameters are located in only one template file; however, some do appear in two or more files. If you are using a parameter that is duplicated in another file, be aware that configuration files are read from left to right and the phone uses the file it reads first.



Troubleshooting: Locating Duplicate Parameters

To check whether a parameter is located in more than one template file, locate the parameter in the reference section [Chapter 13: Configuration Parameters](#).

The table shown next outlines each template file included with UC Software 4.1.0.

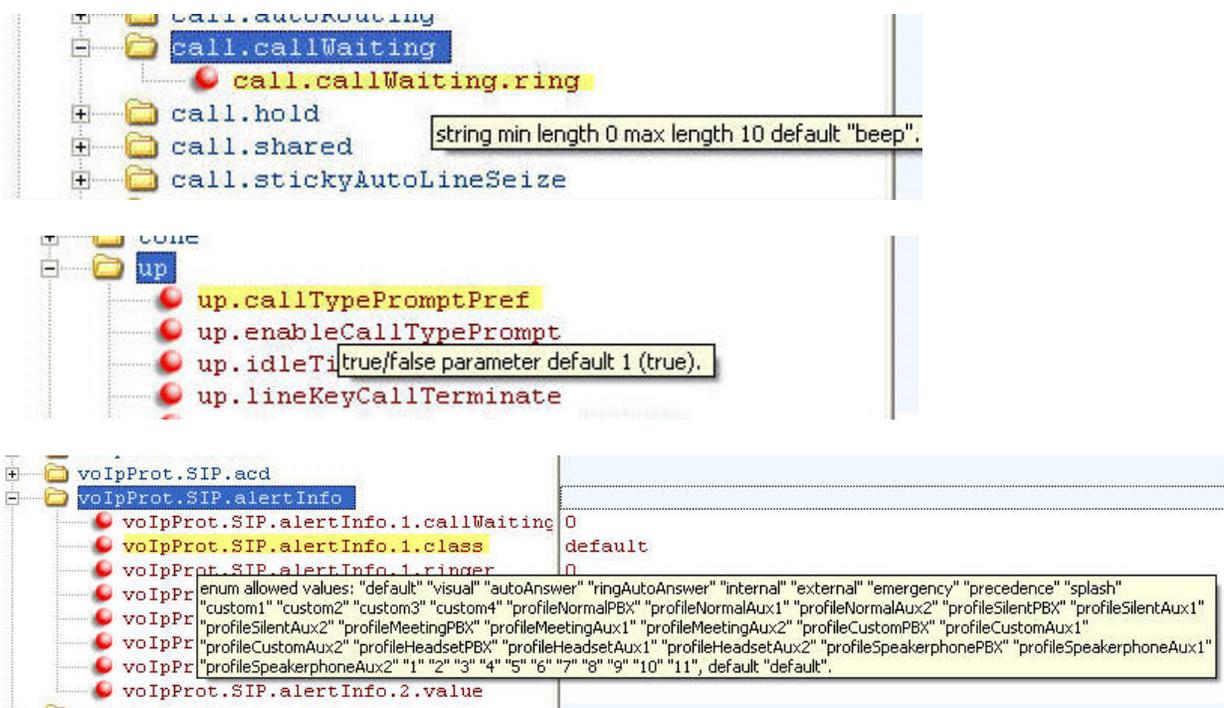
Table 5-1: Configuration File Templates

| Name | Description | Deployment Scenarios |
|------------------|---|---|
| applications.cfg | For applications, browser, microbrowser, XMP-API | Typical Hosted Service Provider
Typical IP-PBX |
| device.cfg | Network Configuration parameters. See Network Interfaces Menu (Ethernet Menu) . | Troubleshooting
Administrative settings |
| features.cfg | Features related enabling corp directory USB recording, CMA, presence, ACD, for example | Typical Hosted Service Provider
Typical IP-PBX |
| firewall-nat.cfg | Firewall parameters | Typical Microsoft Lync Environment |

| <i>Name</i> | <i>Description</i> | <i>Deployment Scenarios</i> |
|------------------|---|--|
| H323.cfg | H.323 video use | Typical Hosted Service Provider using
VVX 1500 for video calls |
| lync.cfg | Microsoft Lync parameters | Typical Microsoft Lync Environment |
| reg-advanced.cfg | Advanced call server, multi-line phones | Typical Hosted Service Provider
Typical IP-PBX |
| reg-basic.cfg | Basic registration | Simple SIP device
Typical Hosted Service Provider |
| region.cfg | Non-North American geographies | Typical Hosted Service Provider
Typical IP-PBX |
| sip-basic.cfg | Basic call server | Simple SIP device
Typical Hosted Service Provider |
| sip-interop.cfg | Advanced call server, multi-line phones | Typical Hosted Service Provider
Typical IP-PBX |
| site.cfg | Multi-site operations | Typical Hosted Service Provider
Typical IP-PBX |
| techsupport.cfg | Available by special request from Polycom Customer Support. | Troubleshooting |
| video.cfg | VVX 1500 video | Typical Hosted Service Provider if using
VVX 1500 for video calls |
| wireless.cfg | SpectraLink parameters specific to Wi-Fi use | |

Along with the templates, UC Software 4.1.0 includes an XML schema file—`polycomConfig.xsd`—that provides information like parameters type (boolean, integer, string, and enumerated type), permitted values, default values, and all valid enumerated type values. View this template file with an XML editor.

A string parameter and a boolean parameter are shown in the following figures.



Changing Configuration Parameter Values

The configuration parameters available in the UC Software use a variety of values, including Boolean, integer, enumerated types, and arrays (a table of values). Each parameter available in the UC Software is listed in alphabetical order in [Chapter 13: Configuration Parameters](#), along with a description, the default value, and the permissible values.

Note that the values for boolean configuration parameters are not case sensitive. The values `0`, `false`, and `off` are inter-changeable and supported. The values `1`, `true`, and `on` are inter-changeable and supported. This Administrators' Guide documents only `0` and `1`.

The following rules apply when you set a parameter with a numeric value outside of its valid range:

- If the configuration file's value is greater than the allowable range, the maximum value is used
- If the configuration file's value is less than the allowable range, the minimum value is used.
- If a parameter's value is invalid, the value is ignored. Invalid parameters values can occur when enumerated type parameters do not match a pre-defined value, when numeric parameters are set to a non-numeric values, when string parameters are either too long or short, or when using null strings in numeric fields. All such situations are logged in the phone's log files.



Tip: Using Blank Values and Special Characters in the Configuration Files

The UC Software interprets 'Null' as empty; that is, `attributeName=""`.

To enter special characters in a configuration file, enter the appropriate sequence using an XML editor:

- & as `&`;
- " as `"`;
- ' as `'`;
- < as `<`;
- > as `>`;
- random numbers as `&0x12;`

Customizing Parameters for a Phone Model

You can customize a set of parameter values for a specific phone model by appending the PHONE MODEL NUMBER descriptor to the parameter. For a list of all phone model names that you can use to create phone-specific configurations, see [Table 12-12: Product Name, Model Name, and Part Number](#). For example:

- `mb.main.home=http://www.myserver.com/index.xhtml`
- `mb.main.home.SPIP560=http://www.myserver.com/ip560.xhtml`
- `mb.main.home.SSIP6000=http://172.24.44.41/`

In this example, all phone models except the SoundPoint IP 560 and SoundStation IP 6000 will use `myserver.com` as the microbrowser home page. The SoundPoint IP 560 will use `ip560.html` and the SoundStation IP 6000 will use the server located at `172.24.44.41/`.

Some configuration parameters cause the phone to reboot or restart when change its value. To find out if a parameter reboots or restarts a phone when changed, locate the parameter in [Chapter 13: Configuration Parameters](#). Parameters that reboot or restart the phone are marked with a superscript (¹ or ²).



Caution: Deprecated Configuration Parameters

Polycom may deprecate configuration parameters that some organizations may still be using – deprecated parameters will not work. To check whether or not you are using deprecated configuration parameters, see the latest Polycom UC Software Release Notes on the [Latest Polycom UC Software Release](#) or check the Release Notes for earlier software versions on the [Polycom UC Software Support Center](#).

Provisioning with the Web Configuration Utility

The Web Configuration Utility enables you to perform configuration changes on a per-phone basis. You can use the Web Configuration Utility as the sole configuration method or in addition to centralized provisioning. If you are provisioning more than ten or twenty phones, Polycom recommends using centralized provisioning as your primary configuration method.



Admin Tip: Updating UC Software on a Single Phone

You can use the Software Upgrade tool in the Web Configuration Utility to update the UC Software version running on a single phone. For detailed information, see [Using the Software Upgrade Tool in the Web Configuration Utility \(Feature Profile 67993\)](#).

Note that configuration changes made to individual phones using the Web Configuration Utility will override configuration settings made with central provisioning. Configuration changes made using a phone's user interface will override settings made using the Web Configuration Utility. If you want to remove changes made using the Web Configuration Utility, click on the **Reset to Default** button on any page in the Web Configuration Utility. This section shows you how to access the Web Configuration Utility.



Web Info: Using the Web Configuration Utility

For more detailed help navigating and using the Web Configuration Utility, see the [Polycom Web Configuration Utility User Guide](#).

You can access the Web Configuration Utility using any of the following Web browsers:

- Microsoft® Internet Explorer 7.0 or later
- Mozilla® Firefox® 3.0.X or later
- Google Chrome™ 10.0.X or later
- Apple® Safari® 5.0.4 or later

Note that as of UC Software 4.0.0, the Web Configuration Utility has been updated, the user interface has been made more user-friendly, and more configuration parameters are available. The Web Configuration Utility comes with built-in contextual help functions that provide you with information and guidance on how to perform basic phone configuration changes. In addition, you can display the interface of the Web Configuration Utility in one of several languages that you can choose once you are logged in.

Accessing the Web Configuration Utility

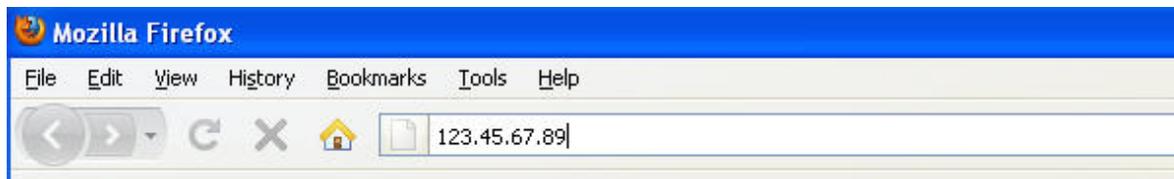
You can access the Web Configuration Utility by entering the phone's IP address in a supported Web browser, for example, `http://<phone IP address>`. If you are a user, log in as **User** –the default password is **123**. If you are an administrator, log in as **Admin** – the default password is **456**.

To access the Web Configuration Utility:

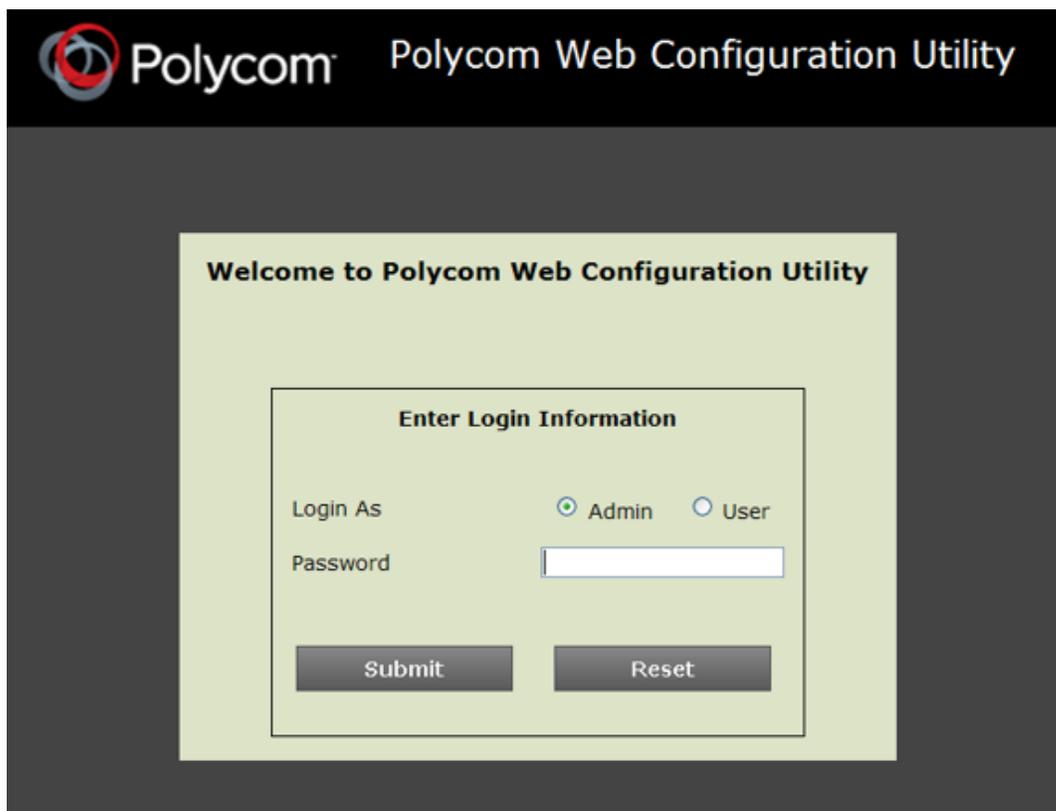
- 1 Select one of the supported Web browsers.

Get your phone's IP address. On the phone, press the **Menu** key and select **Status > Platform > Phone**. Scroll down to see the IP address.

- 2 Enter your phone's IP address in the browser's address bar (as shown next).

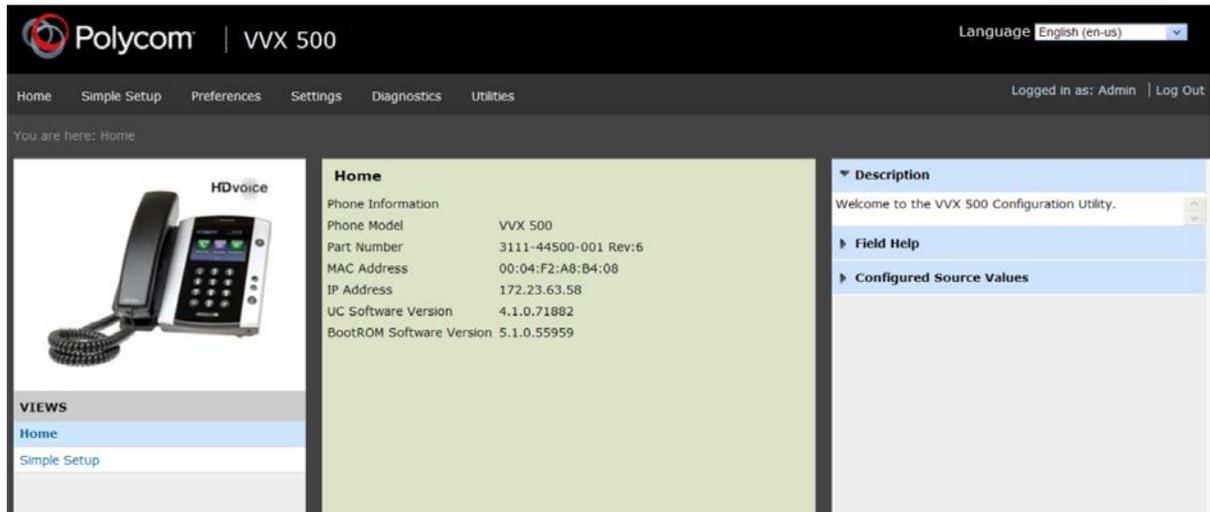


A Web page similar to the one shown next displays.



- 3 Log in as **Admin** – the default administrative password is **456**.

A Web page similar to the one shown next displays.



Use the main menu to navigate through the available settings. The sidebar on the right gives you description of the each page, contextual field help, and the parameter for each setting.

To remove the configuration changes made through the Web Configuration Utility:

- 1 Navigate to the Reset Web Configuration menu on the phone (Menu > Settings > Advanced > Admin Settings > Reset to Defaults > Reset Web Configuration).
- 2 Press the **Yes** soft key.

Your phone may reboot. All Web overrides are removed.



Settings: Some Web Configuration Parameters Do Not Reset

Device.* parameters (for example, device.syslog) that you configure using the Web Configuration Utility will not be saved in the **<MACaddress>-Web.cfg** override file.

Choosing Language Files for the Web Configuration Utility Interface

In the same way you can choose a language for your phone, you can choose a language for viewing the Web Configuration Utility interface. Polycom provides a number of XML language files that you can download from the Polycom UC Software 4.1.0 package to your provisioning server. By default, the SoundPoint IP and IP 321, 331, and 335 phones will display the Web Configuration Utility in English only. If you want these phones to display the Web Utility interface in a language other than English, you will need to copy the corresponding XML language file from the *languages* folder to your provisioning server. This section shows you how to copy the Web Configuration Utility language files to your provisioning server so that phone users can use the Web Configuration Utility interface in the language of their choice.

Certain languages available on Polycom phones use an expanded character set and more memory than other language files. On average, the XML language files for the Web Configuration Utility interface are about 250KB in size. To conserve memory resources, Polycom recommends using only those XML language files for the languages you need. If you want to make multiple languages available to your users, you may need to manage the phone's memory resources. For tips on how to do this, see [Managing the Phone's Memory Resources](#) in *Chapter 11: Troubleshooting Your Polycom Phones*.

To save XML language files to your provisioning server:

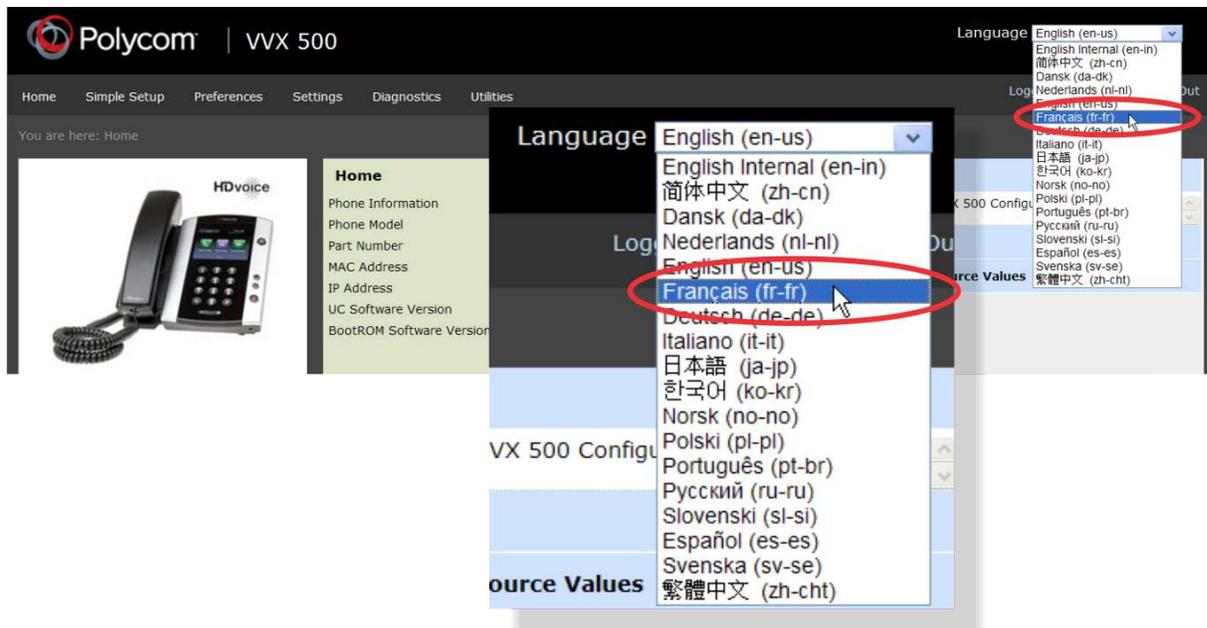
- 1 Create a new folder named *languages* on your provisioning server. This is the folder the provisioning server will read to apply language files to the interface of the Web Configuration Utility. If you need help setting up your provisioning server, see [Chapter 4: Setting Up the Provisioning Server](#).
- 2 Download and unzip the UC software package. You will find all of the language files for the Web Configuration Utility interface in a folder named *languages*.



Note: Don't Confuse Language Files

The *languages* folder located in both the combined and split UC Software versions is not to be confused with the language files for the phone interface, which are located in the *SoundPointIPLocalization* folder. To save memory on the phone, Polycom recommends that you save only the Web Configuration Utility language files that you need to the *languages* folder you created in your provisioning server.

- 3 Copy the XML language file from the *languages* folder you downloaded from the software files to the *languages* folder you created on your provisioning server. For example, if you want the Web Configuration Utility to support French and German, copy `Website_dictionary_language_fr-fr.xml` and `Website_dictionary_language_de-de.xml` to the *languages* folder you created on your provisioning server.
- 4 Login to the Web Configuration Utility and select a language from the Languages drop-down menu at the top-right of the screen, as shown next.



The interface of the Web Configuration Utility displays in the language you select. If the language does not display, ensure that you have extracted and saved the correct language file, or try rebooting the phone.



Troubleshooting: Managing the Phone's Memory Resources

If your selected language will not display, even after you have placed it on the provisioning server and you have rebooted the phone, your phone may have reached its available memory limit. If this occurs, you may need to take steps to manage your phone's available memory resources. For tips on how to manage the phone's memory, refer to [Managing the Phone's Memory Resources](#) in *Chapter 11: Troubleshooting*.

Phone User Interface – Menu System Settings

In addition to centralized provisioning and the Web Configuration Utility, you can use the phone's user interface to change provisioning settings and phone features. As with the Web Configuration Utility, the phone menu system makes some settings available to users and further settings available to administrators. To access administrator settings, such as provisioning values, you will need to enter an administrative password. On the phone user interface, you enter this password in the **Settings** menu. Note that you can use an administrator password where a user password is required, but a user cannot access administrator settings with a user password. The default user password is **123** and the default administrative password is **456**. If you are an administrator and you want to secure the administrative settings from the phone's user interface, change the default administrative password. See [Local User and Administrator Passwords](#) in *Chapter 10: Setting up User and Phone Security Features*.



Settings: Resetting Phone UI Configuration Settings

Configuration made using the phone's user interface will override settings made via the provisioning server configuration files and the Web Configuration Utility. To remove settings made using the phone's user interface, go to **Reset Local Configuration** menu on the phone.



Timesaver: Phone User Interface Menu System

For a map diagram of all menu settings available from the phone user interface, see [Polycom UC Software Menu System](#).

Part III:

Configuring the Phone

Features

Part III describes basic and advanced phones features you can configure for your Polycom® phones. These features include a number of phones features that add efficiency and convenience, audio and video features, and several security features. Chapter 10 gives you a quick overview on how to configure features for single phones once your network system is functioning. This chapter is suitable for administrators or end users.

Before you begin configuring phone features described in this part, take the time to read the short introductory section [Reading the Feature Parameter Tables](#). This section provides important information you need to know in order to successfully perform configuration changes.

Part III consists of the following chapters:

- [Chapter 6: Setting Up Basic Phone Features](#)
- [Chapter 7: Setting Up Advanced Phone Features](#)
- [Chapter 8: Setting Up Phone Audio Features](#)
- [Chapter 9: Setting Up Phone Video Features](#)
- [Chapter 10: Setting Up User and Phone Security Features](#)

Chapter 6: Setting Up Basic Phone Features

After you set up your Polycom® phones with a default configuration on the network, phone users will be able to place and receive calls. However, you may want to add features to the default configuration to suit your organization and user's needs. Polycom provides basic and advanced features that you can configure for the phones to add efficiency and convenience. This chapter will show you how to configure all available basic phone features and call management features.

Before you begin configuring phone features, take the time to read the short introductory section [Reading the Feature Parameter Tables](#). This section provides important information you need to know in order to successfully perform configuration changes.

Basic Phone Features at a Glance

This chapter shows you how to make configuration changes for the following basic features:

- [Configuring the Call Logs](#) Contains call information such as remote party identification, time and date, and call duration in three separate lists, missed calls, received calls, and placed calls.
- [Understanding the Call Timer](#) Maintains a timer, in hours, minutes, and seconds, for each call in progress.
- [Configuring Call Waiting Alerts](#) Visually presents an incoming call on the screen, and plays a configurable sound effect, when you're in another call.
- [Called Party Identification](#) Displays and logs the identity of the party in an outgoing call.
- [Configuring Calling Party Identification](#) Displays a caller's identity, derived from the network signaling, when an incoming call is presented—if the information is provided by the call server.
- [Connected Party Identification](#) Displays and logs the identity of the party to whom you are connected to (if the name is provided by the call server).
- [Distinctive Incoming Call Treatment](#) Automatically applies distinctive treatment to calls containing specific attributes.
- [Applying Distinctive Ringing](#) Enables you to select a ring tone for each line, as well as a ring tone for contacts in the contact directory.
- [Applying Distinctive Call Waiting](#) Enables you to map calls to distinct call waiting types.
- [Configuring Do Not Disturb](#) Temporarily stops incoming calls.

- **Configuring the Handset, Headset, and Speakerphone** SoundPoint IP and VVX phones have a handset and a dedicated headset connection (headset not supplied). All SoundPoint IP, SoundStation IP, and VVX phones have full-duplex speakerphones.
- **Using the Local Contact Directory** The phone maintains a local contact directory that can be downloaded from the provisioning server and edited locally. Any edits to the Contact Directory made on the phone are saved to the provisioning server as a backup.
- **Using the Local Digit Map** The phone has a local set of rules to automate the setup phase of number-only calls.
- **Microphone Mute** Mutes the phone's microphone so other parties cannot hear you. When the microphone mute feature is activated, an icon displays on the phone's screen.
- **Using the Speed Dial Feature** Enables you to place calls quickly from dedicated keys as well as from a speed dial menu.
- **Setting the Time and Date Display** Time and date can be displayed in certain operating modes such as when the phone is idle and during a call.
- **Adding an Idle Display Image** Displays a custom animation on the phone's idle display.
- **Ethernet Switch** Connect your phone to a PC or a LAN.
- **Setting a Graphic Display Background** Enables you to display a picture or graphic on the screen's background.
- **Enabling Multikey Answer** Answer your SpectraLink handset by pressing any key on the keypad.

This chapter also shows you how to make configuration changes for the following basic call management features:

- **Enabling Automatic Off-Hook Call Placement** Supports an optional automatic off-hook call placement feature for each registration.
- **Enabling Call Hold** Pauses activity on one call so that you can use the phone for another task, such as making or receiving another call.
- **Using Call Transfer** Transfers a call in progress to some other destination.
- **Creating Local and Centralized Conferences** You can host or join local conferences or create centralized conferences using conference bridge numbers. The advanced aspects of conferencing, like managing parties, are part of the Productivity Suite.
- **Enabling Conference Management** Add, hold, mute, and remove conference participants, and obtain information about participants.
- **Configuring Call Forwarding** Provides a flexible call forwarding feature to forward calls to another destination.
- **Configuring Directed Call Pick-Up and Enabling Group Call Pickup** Enables you to pick up calls to another phone by dialing the extension of the other phone. Calls to another phone within a pre-defined group can be picked up without dialing the extension of the other phone.

- [Configuring Call Park and Retrieve](#) Park an active call—puts it on hold to a specific location, so it can be retrieved by any phone.
- [Enabling Last Call Return](#) Automatically redials the number of the last received call.

To troubleshoot any problems with your Polycom phones on the network, see [Troubleshooting Your Polycom Phones](#). For more information on the Web Configuration Utility, see [Provisioning with the Web Configuration Utility](#). For instructions on how to read the feature descriptions in this section, see [Reading the Feature Parameter Tables](#).

Configuring the Call Logs

The phone records and maintains phone events to a call log, also known as a call list. These call logs contain call information such as remote party identification, time and date of the call, and call duration. The log is stored as a file in XML format named **<MACAddress>-calls.xml** to your provisioning server. If you want to route the call logs to another server, use the CALL_LISTS_DIRECTORY field in the master configuration file. You can use the call logs to redial previous outgoing calls, return incoming calls, and save contact information from call log entries to the contact directory. All call logs are enabled by default. See [Table 6-1: Configuring the Call Logs](#) for instructions on how to enable or disable the call logs.

The phones automatically maintain the call logs in three separate call lists: Missed Calls, Received Calls, and Placed Calls. Each of these call lists can be cleared manually by individual phone users. You can delete individual records or all records in a group (for example, all missed calls). You can also sort the records or filter them by line registration.

The call lists on the SoundPoint IP and SoundStation IP phones will not be cleared or deleted when the phone reboots. As of Polycom UC Software 4.0.1, the VVX 500 and 1500 phones and SpectraLink handsets will remember the previous call history after a restart or reboot.



Tip: Merged Call Lists

On some phones, missed and received calls will display in one call list. In these combined lists, you can identify call types by the icons:

Missed call icon  Received call icon 

Table 6-1: Configuring the Call Logs

| Central Provisioning Server | template > parameter |
|--|---|
| Enable or disable call logs or individual call logs..... | features.cfg > feature.callList.enabled |
| Enable or disable the missed call list..... | features.cfg > feature.callListMissed.enabled |
| Enable or disable the placed call list..... | features.cfg > feature.callListPlaced.enabled |
| Enable or disable the received call list..... | features.cfg > feature.callListReceived.enabled |

Example Call Log Configuration

The following illustration shows you each of the call log parameters you can enable or disable in the **features.cfg** template file.

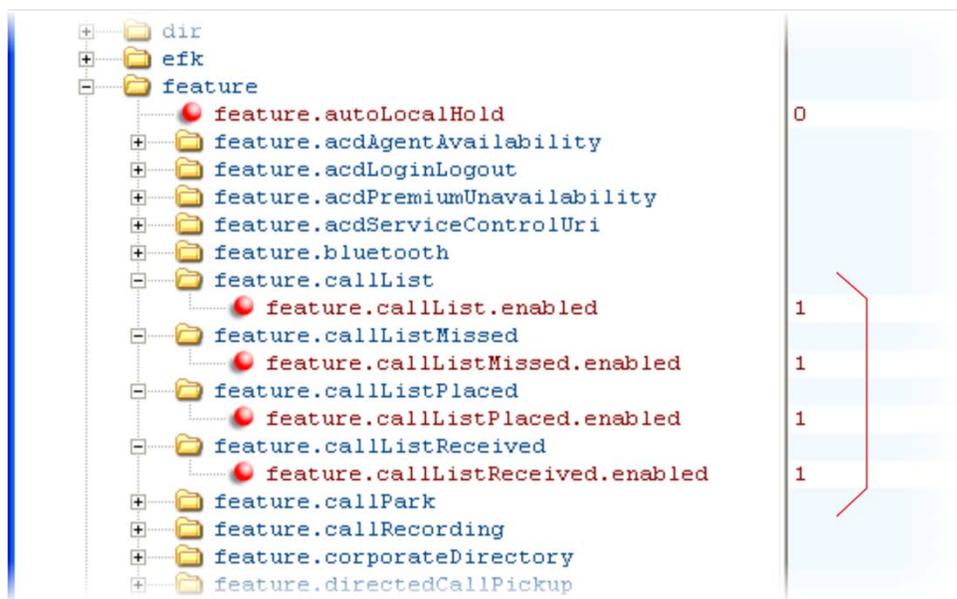


Table 6-2: Call Log Elements and Attributes describes each element and attribute that displays in the call log. Polycom recommends using an XML editor such as XML Notepad 2007 to view and edit the call log. Note that you can place the elements and attributes in any order in your configuration file.

Table 6-2: Call Log Elements and Attributes

| <i>Element</i> | <i>Permitted Values</i> |
|---|--|
| direction
Call direction with respect to the user. | In, Out |
| disposition
What happened to the call. When a call entry is first created, the disposition is set to Partial. | Busy, Forwarded, Normal, Partial, Preempted, Rejected, RemotelyHandled, Transferred |
| line
The line (or registration) index. | Positive integer |
| protocol
The line protocol. | SIP or H323 |

| <i>Element</i> | <i>Permitted Values</i> |
|---|-------------------------|
| startTime | String |
| The start time of the call. For example: 2010-01-05T12:38:05 in local time. | |
| duration | String |
| The duration of the call, beginning when it is connected and ending when the call is terminated. For example: PT1H10M59S. | |
| count | Positive Integer |
| The number of consecutive missed and abandoned calls from a call destination. | |
| destination | Address |
| 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). | |
| source | Address |
| The source of the call (caller ID from the call recipient's perspective). | |
| Connection | Address |
| 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. | |
| finalDestination | Address |
| The final connected party of a call that has been forwarded or transferred to a third party. | |

Understanding the Call Timer

A call timer displays on the phone's screen. A separate call duration timer displays the hours, minutes, and seconds of each call in progress.

There are no related configuration changes.

Configuring Call Waiting Alerts

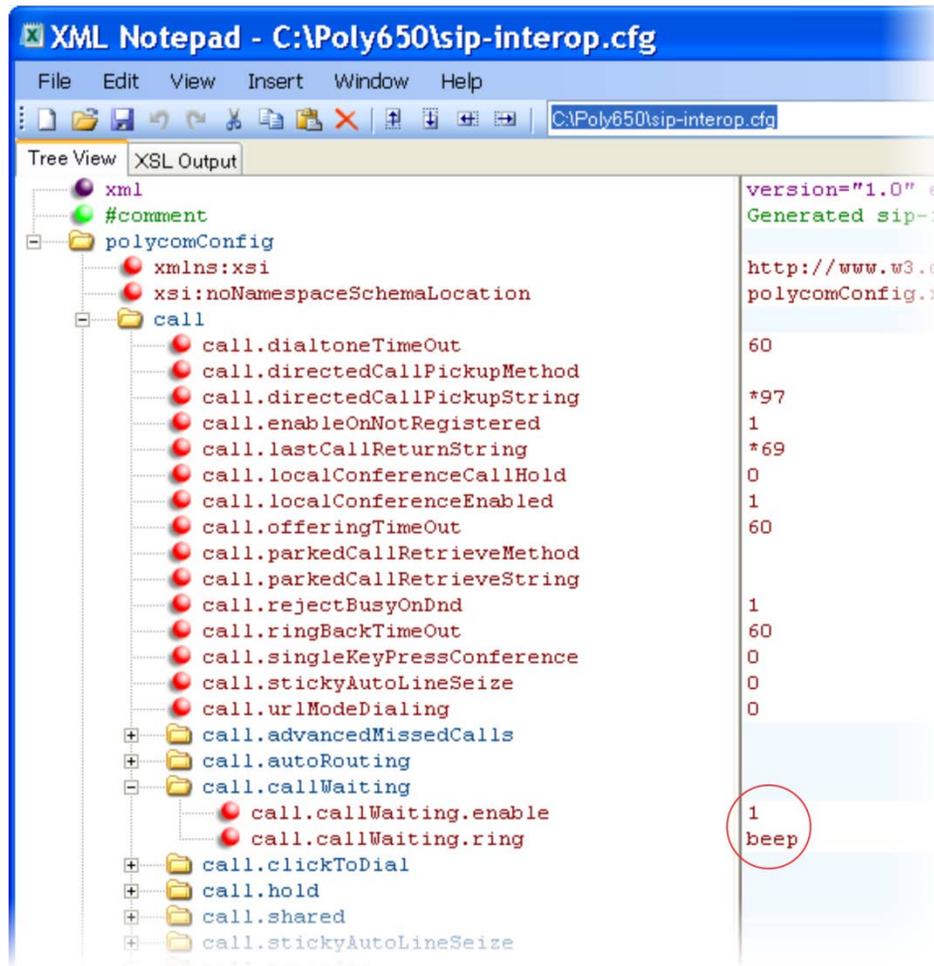
By default, the phone will alert you to incoming calls while you are in an active call. As shown in [Table 6-3: Configuring Call Waiting Alerts](#), you can disable call waiting alerts and you can specify the ringtone of incoming calls.

Table 6-3: Configuring Call Waiting Alerts

| Central Provisioning Server | template > parameter |
|---|---|
| Enable or disable call waiting..... | sip-interop.cfg > call.callWaiting.enable |
| Specify the ringtone of incoming calls when you are in an active call | sip-interop.cfg
> call.callWaiting.ring |

Example Call Waiting Configuration

The following illustration shows you where to disable call waiting alerts and how to change the ringtone of incoming calls in the **sip-interop.cfg** template.



Called Party Identification

By default, the phone displays and logs the identity of parties called from the phone. The phone obtains called party identity from the network signaling. Because Called Party Identification is a default state, the phone will display caller IDs matched to the call server and does not match IDs to entries in the Local Contact Directory or Corporate Directory.

There are no related configuration changes.

Configuring 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 choose to display the name you assigned there, as shown in [Table 6-4: Configuring Calling Party Identification](#). Note that the phone cannot match the identity of calling parties to entries in the Corporate Directory.



Note: Automatic Caller ID Scrolling on SoundPoint IP 321, 331, and 335 Phones

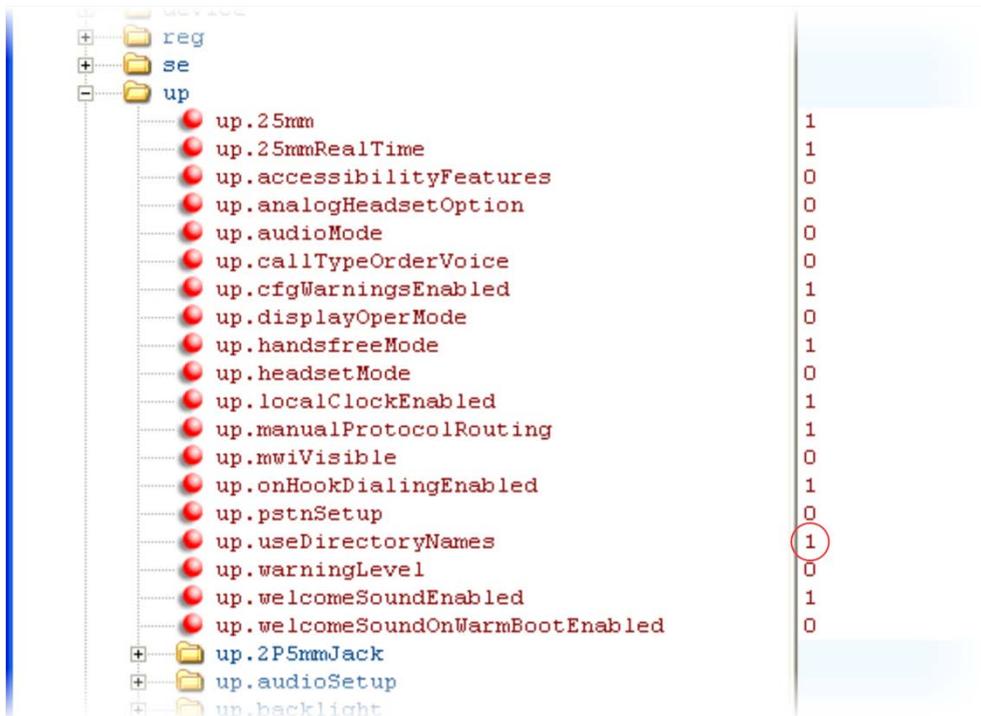
As of Polycom® UC Software 3.3.0, when the SoundPoint IP 321, 331, and 335 phones receive incoming calls, the caller ID will automatically scroll from left to right. Auto-scrolling stops once the call is connected, but you can use the left and right arrow keys to scroll manually.

Table 6-4: Configuring Calling Party Identification

| | |
|--|--------------------------------|
| Central Provisioning Server | template > parameter |
| Substitute the network address ID with the Contact Directory name reg-advanced.cfg
> up.useDirectoryNames | |
| Web Configuration Utility | |
| Specify whether or not to substitute the network address with the Contact Directory name. Navigate to Preferences > Additional Preferences > User Preferences . | |

Example Calling Party Configuration

The following illustration shows you how to substitute the network address caller ID with the name you assigned to that contact in the contact directory. The ID of incoming call parties will display on the phone screen.



Configuring PSTN Calling Party Identification

The SoundStation Duo conference phone is the only Polycom phone running Polycom UC Software that supports PSTN mode. This section applies to SoundStation Duo conference phones only.

Caller ID, the display of an incoming caller's information on the phone, is a subscription service with standards that vary by country. Check with your local telephone service provider to determine if this service is available in your area (British Telecom and Japanese caller ID standards are not supported). If the service is available, you will need to configure two basic settings before the SoundStation Duo can use the caller ID standard in use for your country. For information on how to configure the two basic settings, see [PSTN Communications Settings](#).

Use the following table as a guideline for choosing the correct caller ID standard. If you need further information, consult your telephone service provider.

Table 6-5: PSTN Caller ID Standards

| <i>Country</i> | <i>Caller ID Standard</i> |
|---|---------------------------|
| USA, Hong Kong, Singapore, Canada | Bellcore |
| Austria, Belgium, France, Germany, Luxemburg, Norway, Poland, Spain, Czech Republic, Slovenia, Switzerland, Taiwan, Turkey, South Africa, Italy | ETSI |
| China, Denmark, Finland, Greece, Netherlands, Portugal, Sweden, Uruguay, Brazil | DTMF |



Note: UC Software 4.0.1 does not support the British Telecom caller ID standard.

The British Telecom and Japanese Caller ID standards are not supported.

Enabling Missed Call Notification

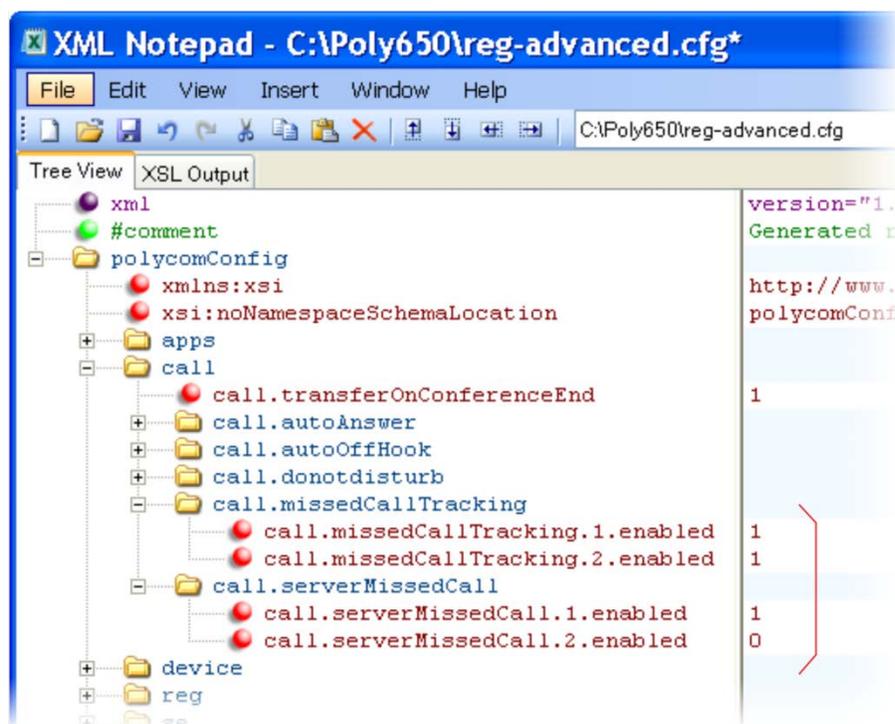
You can display on the phone's screen a counter that shows the number of missed calls. To reset the counter, view the Missed Calls list on the phone. As [Table 6-6: Enabling Missed Call Notification](#) indicates, you can also configure the phone to record all missed calls or to display only missed calls that arrive through the Session Initiation Protocol (SIP) server. You can enable Missed Call Notification for each registered line on a phone.

Table 6-6: Enabling Missed Call Notification

| Central Provisioning Server | template > parameter |
|--|----------------------|
| Enable or disable the missed call counter for a specific registration
..... reg-advanced.cfg > call.missedCallTracking.x.enabled | |
| Specify, on a per-registration basis, whether to display all missed calls or only server-generated missed calls
..... reg-advanced.cfg > call.serverMissedCall.x.enabled | |

Example Missed Call Notification Configuration

In the following example, the missed call counter is enabled by default for registered lines 1 and 2, and only server-generated missed calls will be displayed on line 1.



Connected Party Identification

By default, the phone displays and logs the identity of remote parties you connect to if the call server can derive the name and ID from the network signaling. Note that 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. For example, Bob places a call to Alice, but Alice has call diversion configured to divert Bob's incoming calls to Fred. In this

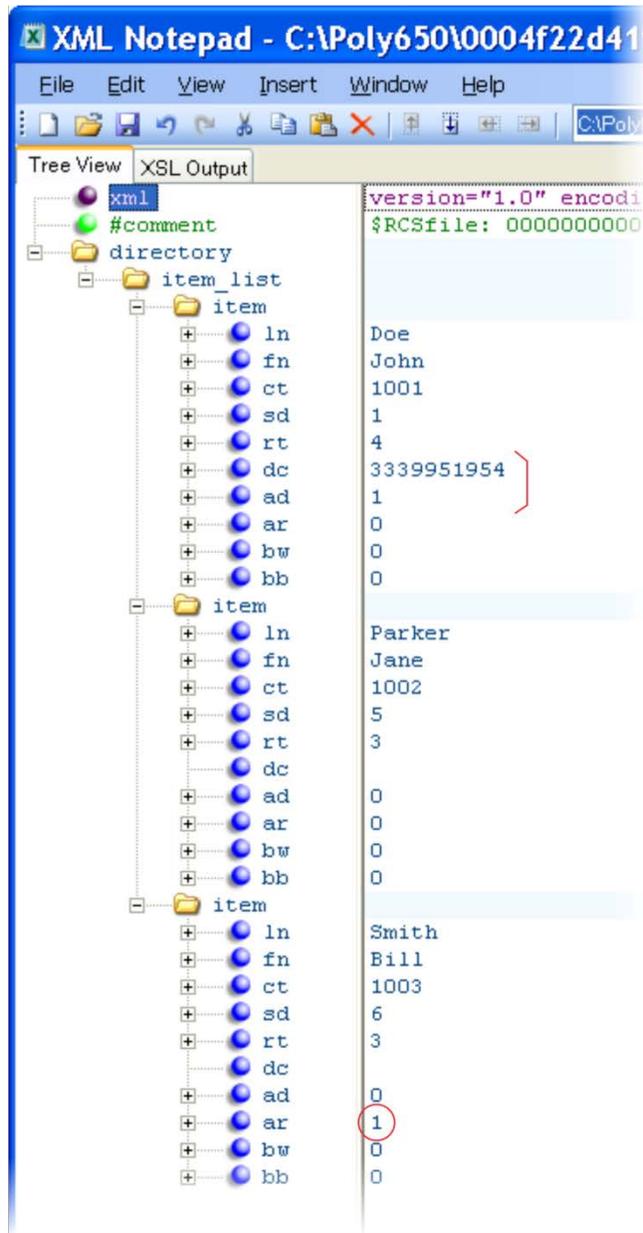
case, the phone will log and display the connection between Bob and Fred. Note that the phone does not match party IDs to entries in the contact directory or the corporate directory.

Distinctive Incoming Call Treatment

You can apply distinctive treatment to specific calls and contacts in your contact directory. You can set up distinctive treatment for each of your contacts by specifying a **Divert Contact**, enabling **Auto-Reject**, or by enabling **Auto-Divert** for a specific contact in the local contact directory (see [Using the Local Contact Directory](#)). You can also apply distinctive treatment to calls and contacts through the phone's user interface.

Example Call Treatment Configuration

In the following example, the Auto Divert feature has been enabled in `ad` so that incoming calls from John Doe will be diverted to SIP address `3339951954` as specified in `dc`. Incoming calls from Bill Smith have been set to Auto Reject in `ar` and will be sent to voicemail.



Note that if you enable both the Auto Divert and Auto Reject features, Auto Divert has precedence over Auto Reject. For a list of all parameters you can use in the contact directory, see [Table 6-12: Understanding the Local Contact Directory](#).

Applying Distinctive Ringing

The distinctive ringing feature enables you to apply a distinctive ringtone to a registered line, a specific contact, or type of call.

There are three ways to set distinctive ringing and [Table 6-7: Applying Distinctive Ringing](#) shows you the parameters for each. If you set up distinctive ringing using more than one of the following methods, the phone will use the highest priority method.

- You can assign ringtones to specific contacts in the Contact Directory. For more information, see [Distinctive Incoming Call Treatment](#). This option is first and highest in priority.
- You can use the `voIpProt.SIP.alertInfo.x.value` and `voIpProt.SIP.alertInfo.x.class` parameters in the **sip-interop.cfg** template 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.
- You can select a ringtone for each registered line on the phone. Press the **Menu** key, and select **Settings > Basic > Ring Type**. This option has the lowest priority.

Table 6-7: Applying Distinctive Ringing

| Central Provisioning Server | template > parameter |
|---|--|
| Map alert info string in the SIP header to ringtones..... | sip-interop.cfg > <code>volpProt.SIP.alertInfo.x.class</code> |
| | sip-interop.cfg > <code>volpProt.SIP.alertInfo.x.value</code> |
| Set default profiles and ringtones | wireless.cfg > <code>np*</code> |
| Specify a ringtone for a specific registered line | reg-advanced.cfg > <code>reg.x.ringType</code> |
| Specify ringtones for contact directory entries | 000000000000-directory~.xml |

Local Phone User Interface

You can edit the ringtone of each registered line by navigating to **Menu > Settings > Basic > Ring Type**.

To edit the ringtone for a specific contact, navigate to **Menu > Features > Contact Directory**, highlight a contact, press the **Edit** soft key, and specify a value for the **Ring Type**.

Example Distinctive Ringing Configuration

The following illustration shows that the ring type `ringer2` has been applied to incoming calls to line 1.



For a list of all parameters and their corresponding ringtones, see [Table 13-76: Ringtone Pattern Names](#).

Applying 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. [Table 6-8: Applying Distinctive Call Waiting](#) shows you the parameters you can configure for this feature. This feature requires call server support.

Table 6-8: Applying Distinctive Call Waiting

| Central Provisioning Server | template > parameter |
|--|--|
| Enter the string which displays in the SIP alert-info header | sip-interop.cg |
| > volpProt.SIP.alertInfo.x.value | |
| Enter the ring class name | sip-interop.cfg > volpProt.SIP.alertInfo.x.class |

Example Distinctive Call Waiting Configuration

In the following illustration, `voIpProt.SIP.alertInfo.1.value` is set to `http://<SIP headerinfo>`. An incoming call with this value in the SIP alert-info header will cause the phone to ring in a manner specified by `voIpProt.SIP.alertInfo.x.class`. In this example, the phone will display a visual LED notification, as specified by the value `visual`.

The screenshot shows a configuration tree for `voIpProt.SIP`. The tree is expanded to show the `voIpProt.SIP.alertInfo` folder, which contains three sub-items: `voIpProt.SIP.alertInfo.1.class`, `voIpProt.SIP.alertInfo.1.value`, `voIpProt.SIP.alertInfo.2.class`, and `voIpProt.SIP.alertInfo.2.value`. The `voIpProt.SIP.alertInfo.1.value` is set to `http://<SIPheaderinfo>` and the `voIpProt.SIP.alertInfo.1.class` is set to `visual`. A red bracket on the right side of the screenshot groups these two values together.

| Parameter | Value |
|---|------------------------|
| <code>voIpProt.SIP.acceptMissingVideoFmtp</code> | 1 |
| <code>voIpProt.SIP.allowTransferOnProceeding</code> | 1 |
| <code>voIpProt.SIP.authOptimizedInFailover</code> | 0 |
| <code>voIpProt.SIP.csta</code> | 0 |
| <code>voIpProt.SIP.failoverOn503Response</code> | 1 |
| <code>voIpProt.SIP.lcs</code> | 0 |
| <code>voIpProt.SIP.ms-forking</code> | 0 |
| <code>voIpProt.SIP.pingInterval</code> | 0 |
| <code>voIpProt.SIP.pingMethod</code> | PING |
| <code>voIpProt.SIP.sendCompactHdrs</code> | 0 |
| <code>voIpProt.SIP.strictLineSeize</code> | 0 |
| <code>voIpProt.SIP.strictReplacesHeader</code> | 1 |
| <code>voIpProt.SIP.strictUserValidation</code> | 0 |
| <code>voIpProt.SIP.tcpFastFailover</code> | 0 |
| <code>voIpProt.SIP.turnOffNonSecureTransport</code> | 0 |
| <code>voIpProt.SIP.use486forReject</code> | 0 |
| <code>voIpProt.SIP.useCompleteUriForRetrieve</code> | 1 |
| <code>voIpProt.SIP.useContactInReferTo</code> | 0 |
| <code>voIpProt.SIP.useRFC2543hold</code> | 0 |
| <code>voIpProt.SIP.useSendonlyHold</code> | 1 |
| <code>voIpProt.SIP.WM50</code> | 0 |
| <code>voIpProt.SIP.acd</code> | |
| <code>voIpProt.SIP.alertInfo</code> | |
| <code>voIpProt.SIP.alertInfo.1.class</code> | visual |
| <code>voIpProt.SIP.alertInfo.1.value</code> | http://<SIPheaderinfo> |
| <code>voIpProt.SIP.alertInfo.2.class</code> | default |
| <code>voIpProt.SIP.alertInfo.2.value</code> | |
| <code>voIpProt.SIP.assuredService</code> | |
| <code>voIpProt.SIP.CID</code> | |
| <code>voIpProt.SIP.compliance</code> | |

Configuring Do Not Disturb

You can use the Do Not Disturb (DND) feature to temporarily stop incoming calls. You can also turn off audio alerts and receive visual call alerts only, or you can make your phone appear busy to incoming callers. Incoming calls received while DND is turned on are logged as missed.

DND can be enabled locally through the phone or through a server. [Table 6-9: Configuring Do Not Disturb](#) lists parameters for both methods. The local DND feature is enabled by default, and you have the option of disabling it. When local DND is enabled, you can turn DND on and off using the **Do Not Disturb** button on the phone. Local DND can be configured only on a per-registration basis. If you want to forward calls while DND is enabled, see [Configuring Call Forwarding](#).



Note: Using Do Not Disturb on Shared Lines

A phone that has DND enabled and activated on a shared line will visually alert you to an incoming call, but the phone will not ring.

If you want to enable server-based DND, you must enable the feature on both a registered phone and on the server. The benefit of server-based DND is that if a phone has multiple registered lines, you can apply DND to all line registrations on the phone; however, you cannot apply DND to individual registrations on a phone that has multiple registered lines. Note that although server-based DND disables the local Call Forward and DND features, if an incoming is not routed through the server, you will still receive an audio alert.

Server-based DND behaves the same way as the pre-SIP 2.1 per-registration feature with the following exceptions:

- You cannot enable server-based DND if the phone is configured as a shared line.
- If server-based DND is enabled but not turned on, and you press the DND key or select DND on the phone's Features menu, the 'Do Not Disturb' message will display on the phone and incoming calls will continue to ring.

Table 6-9: Configuring Do Not Disturb

| Central Provisioning Server | template > parameter |
|---|--|
| Enable or disable server-based DND | sip-interop.cfg > volpProt.SIP.serverFeatureControl.dnd |
| Enable or disable local DND behavior when server-based enabled
..... | sip-interop.cfg > volpProt.SIP.serverFeatureControl.localProcessing.dnd |
| Specify whether, when DND is turned on, the phone rejects incoming calls with a busy signal or gives you a visual and no audio alert. | sip-interop.cfg > call.rejectBusyOnDnd |
| Enable DND as a per-registration feature or use it as a global feature for all registrations
..... | reg-advanced.cfg > call.donotdisturb.perReg |
| <hr/> | |
| Local Phone User Interface | |
| If DND is enabled, you can turn DND on or off using the Do Not Disturb key on the SoundPoint IP 550, 560, and 650, and the VVX 500 and 1500 or the Do Not Disturb menu option in the Features menu on the SoundPoint IP 321, 331, 335 and 450, the SoundStation IP 5000, and 6000, and the SpectraLink handsets. | |

Example Do Not Disturb Configuration

In the following example, taken from the `sip-interop.cfg` template, server-based DND has been enabled in `serverFeatureControl.dnd`, and `rejectBusyOnDnd` has been set to 1 – enabled – so that when you turn on DND on the phone, incoming callers will receive a busy signal.

```

version="1.0"
Generated sip-...
http://www.w3.c...
polycomConfig...

60
*97
1
*69
0
1
60
1
60
0

voIpProt.SIP.requestURI
voIpProt.SIP.requestValidation
voIpProt.SIP.serverFeatureControl
  voIpProt.SIP.serverFeatureControl.cf 0
  voIpProt.SIP.serverFeatureControl.dnd 1
  voIpProt.SIP.serverFeatureControl.missedCalls 0
  voIpProt.SIP.serverFeatureControl.localProcessing
    voIpProt.SIP.serverFeatureControl.localProcessing.cf 1
    voIpProt.SIP.serverFeatureControl.localProcessing.dnd 1
voIpProt.SIP.specialEvent
  
```



Note: DND LED Alerts on the VVX

The LED on the Do Not Disturb key on the VVX 1500 is red when pressed or when server-based DND is enabled.

Configuring the Handset, Headset, and Speakerphone

All SoundPoint IP and VVX phones come with a handset and a dedicated connector for a headset and include support for a USB headset; all Polycom phones have built-in speakerphones. You can enable and disable each of these options, as shown in [Table 6-10: Configuring the Handset, Headset, and Speakerphone](#). Note that although handsets are shipped with your phones, headsets are not provided.

SoundPoint IP and VVX phones have a dedicated key to switch between speakerphone and headset. You can enable or disable the handsfree speakerphone mode. SpectraLink headsets support Bluetooth v2.1 headsets with Enhanced Data Rate (EDR) and Headset Profile (HSP v1.2).



Web Info: Configuring an External Electronic Hookswitch

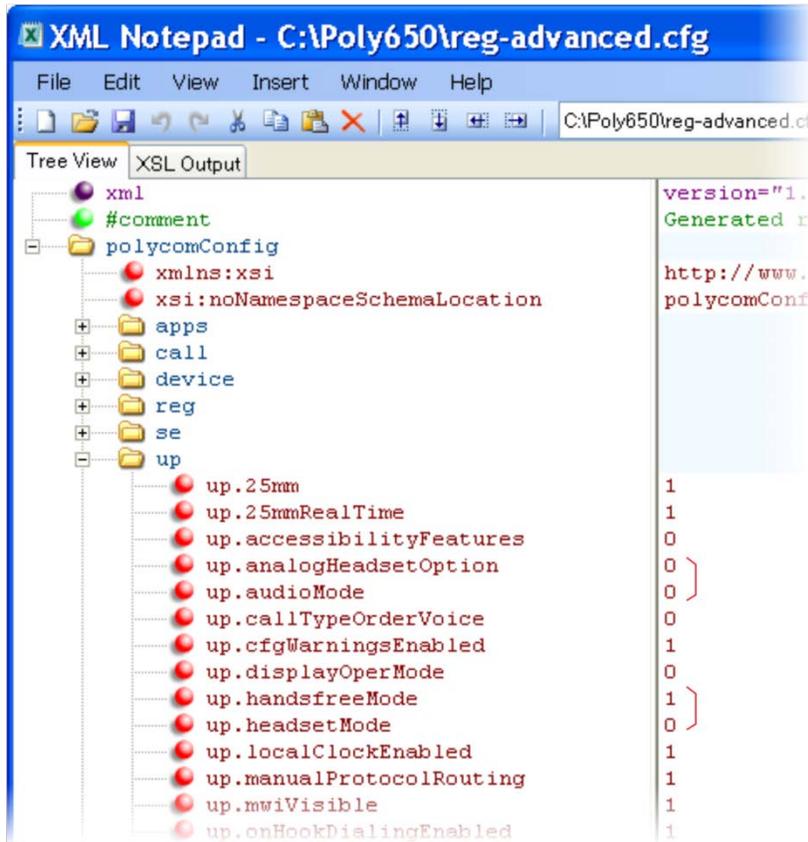
You can configure all supported Polycom desktop phones with an external electronic hookswitch. For more information, see [Technical Bulletin 35150: Using an Electronic Hookswitch with SoundPoint IP and Polycom VVX 1500 Phones](#).

Table 6-10: Configuring the Handset, Headset, and Speakerphone

| Central Provisioning Server | template > parameter |
|---|--|
| Enable or disable headset memory mode | reg-advanced.cfg and site.cfg > up.headsetMode |
| Enable or disable handsfree speakerphone mode | reg-advanced.cfg and site.cfg > up.handsfreeMode |
| Specify if the electronic hookswitch is enabled and what type of headset is attached | reg-advanced.cfg and site.cfg > up.analogHeadsetOption |
| Specify if the handset or a headset should be used for audio | reg-advanced.cfg and site.cfg > up.audioMode |
| Turn the SpectraLink Bluetooth radio on or off (must be turned on to use a Bluetooth headset) | features.cfg > bluetooth.radioOn |
| Specify how phone and the USB headset interact | site.cfg > up.headset.phoneVolumeControl |
| Specify if the USB headset volume persists between calls | site.cfg > voice.volume.persist.headset |
| <hr/> | |
| Web Configuration Utility | |
| To enable or disable headset memory mode, navigate to Preferences > Additional Preferences > User Preferences . | |
| <hr/> | |
| Local Phone User Interface | |
| To enable or disable headset memory mode, navigate to Settings > Basic > Preferences > Headset > Headset Memory Mode | |
| To enable or disable handsfree speakerphone mode navigate to Settings > Advanced > Admin Settings > Handsfree Mode . | |

Example Handset, Headset, and Speakerphone Configuration

The following illustration shows the default settings in the **reg-advanced.cfg** template. In this example, handsfree mode is enabled and headset memory mode and electronic hookswitch are disabled.



Using the Local Contact Directory

The phones feature a contact directory you can use to store frequently used contacts.

Note that the phone follows a precedence order when looking for a contact directory. A phone will look first for a local directory in its own memory, next for a **<MACaddress>-directory.xml** that is uploaded to the server, and finally for a seed directory **000000000000-directory~.xml** that is included in your UC software download.

Changes you make to the contact directory from the phone are stored on the phone drive and uploaded to the provisioning server in **<MACaddress>-directory.xml**. This enables you to preserve a contact directory during reboots.

If you want to use the seed directory, locate **000000000000-directory~.xml** in your UC Software files on the server and remove the tilde (~) from the file name. The phone will substitute its own MAC address for **<000000000000>**.

The contact directory is the central database for several phone features including speed dial (see [Using the Speed Dial Feature](#)), distinctive incoming call treatment (see [Distinctive Incoming Call Treatment](#)), presence (see [Using the Presence Feature](#)), and instant messaging (see [Enabling Instant Messaging](#)). [Table 6-11: Using the Local Contact Directory](#), shown next, lists the directory parameters you can configure. The SoundPoint IP and SoundStation IP phones support up to 99 contacts, while the VVX phones and SpectraLink handsets support up to 999 contacts. If you want to conserve phone memory, you can configure the phones to support a lower maximum number of contacts.



Tip: Deleting the Per-Phone Contact Directory

If you created a per-phone `<MACaddress>directory.xml` for a phone and you want that phone to use a global contact directory `000000000000-directory.xml`, remove the `<MACaddress>directory.xml` file you created from the server.

Table 6-11: Using the Local Contact Directory

| Central Provisioning Server | template > parameter |
|--|---|
| Enable or disable the local contact directory | <code>features.cfg>feature.directory.enabled</code> |
| Specify if the local contact directory is read-only | <code>features.cfg > dir.local.readonly</code> |
| Specify the maximum number of contact entries for each phone
..... | <code>features.cfg> dir.local.contacts.maxNum</code> |
| Specify whether to search the directory by first name or last name | <code>features.cfg > dir.search.field</code> |
| The template contact directory file | <code>000000000000-directory~.xml</code> |
| Local Phone User Interface | |
| To edit the contact directory on the phone, navigate to Features > Contact Directory , choose a contact, and press the Edit soft key. | |

Example Configuration

The following illustration shows four contacts configured in a directory file.

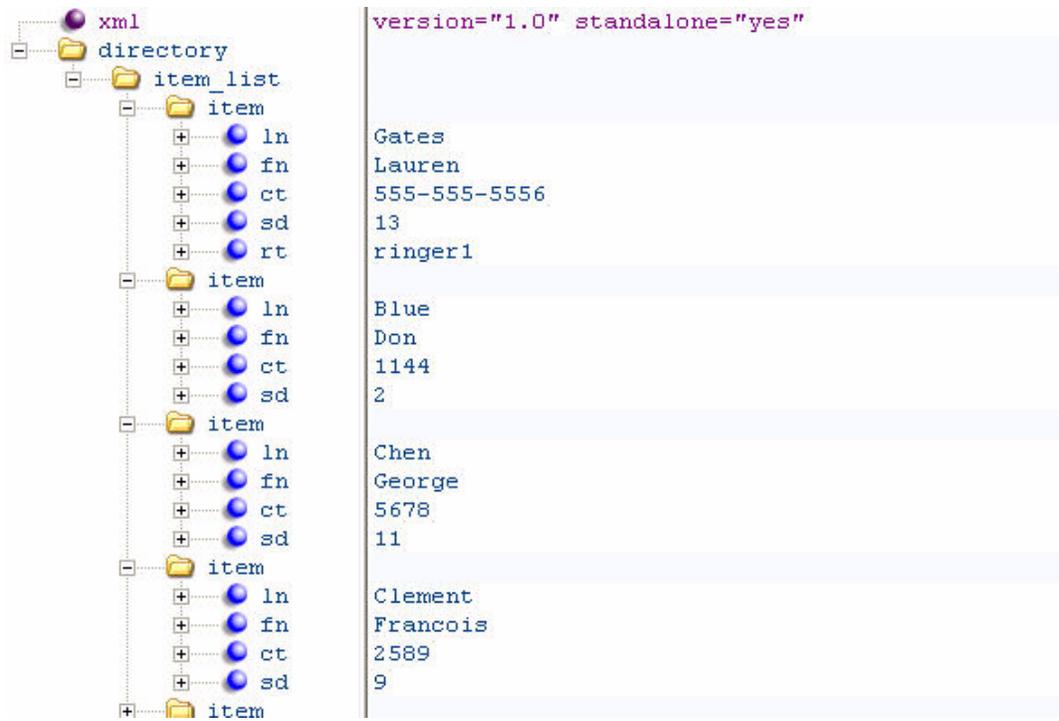


Table 6-12: Understanding the Local Contact Directory, shown next, describes each of the parameter elements and permitted values that you can use in the local contact directory.

Table 6-12: Understanding the Local Contact Directory

| Element | Definition | Permitted Values |
|-----------|---|--|
| fn | First Name
The contact's first name. | UTF-8 encoded string of up to 40 bytes ¹ |
| ln | Last Name
The contact's last name. | UTF-8 encoded string of up to 40 bytes ¹ |
| ct | Contact
Used by the phone to address a remote party in the same way that a string of digits or a SIP URL are dialed manually by the user. This element is also used to associate incoming callers with a particular directory entry. The maximum field length is 128 characters.
<i>Note:</i> This field cannot be null or duplicated. | UTF-8 encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL |
| sd | Speed Dial Index
Associates a particular entry with a speed dial key for one-touch dialing or dialing from the speed dial menu.
<i>Note:</i> On SoundPoint IP 321, 331, and 335 phones, the maximum speed-dial index is 99. | Null, 1 to 9999 |

| <i>Element</i> | <i>Definition</i> | <i>Permitted Values</i> |
|----------------|--|---|
| lb | Label | UTF-8 encoded string of up to 40 bytes¹ |
| | The label for the contact. <i>Note:</i> 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 will form the label. A space is added between first and last names. | |
| pt | Protocol | SIP, H323, or Unspecified |
| | The protocol to use when placing a call to this contact. | |
| rt | Ring Tone | Null, 1 to 21 |
| | When incoming calls match a directory entry, this field specifies the ringtone that will be used. | |
| dc | Divert Contact | UTF-8 encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL |
| | The address to forward calls to if the Auto Divert feature is enabled. | |
| ad | Auto Divert | 0 or 1 |
| | If set to 1, callers that match the directory entry are diverted to the address specified for the divert contact element.
<i>Note:</i> If auto-divert is enabled, it has precedence over auto-reject. | |
| ar | Auto Reject | 0 or 1 |
| | If set to 1, callers that match the directory entry specified for the auto-reject element are rejected.
<i>Note:</i> If auto divert is also enabled, it has precedence over auto reject. | |
| bw | Buddy Watching | 0 or 1 |
| | If set to 1, this contact is added to the list of watched phones. | |
| bb | Buddy Block | 0 or 1 |
| | If set to 1, this contact is blocked from watching this phone. | |

¹ In some cases, this will be less than 40 characters due to UTF-8's variable bit length encoding.

Using the Local Digit Map

The phone has a local digit map feature that, when configured, will automatically call a dialed number, eliminating the need to press the **Dial** or **Send** soft key to place outgoing calls. Note that digit maps do not apply to on-hook dialing.

Digit maps are defined by a single string or a list of strings. If a number you dial matches any string of a digit map, the call is automatically placed. If a number you dial 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 will be placed. The parameter for

each of these options is outlined in [Table 6-13: Using the Local Digit Map](#). The configuration syntax of the digit map is based on recommendations in section 2.1.5 of [RFC 3435](#).



Web Info: Changing the Local Digit Map on Polycom Phones

For instructions on how to modify the Local Digit Map, see [Technical Bulletin 11572: Changes to Local Digit Maps on SoundPoint IP, SoundStation IP, and Polycom VVX 1500 Phones](#).

Table 6-13: Using the Local Digit Map

| Central Provisioning Server | template > parameter |
|--|--|
| Apply a dial plan to dialing scenarios..... | site.cfg > dialplan.applyTo* |
| Specify the digit map to use for the dial plan | site.cfg > dialplan.digitmap |
| Specify the timeout for each segment of the digit map..... | site.cfg > dialplan.digitmap.timeOut |
| Specify the behavior if an impossible dial plan match occurs..... | site.cfg > dialplan.impossibleMatchHandling |
| Specify if trailing # digits should be removed from digits sent out .. | site.cfg > dialplan.removeEndOfDial |
| Specify the details for emergency dial plan routing | site.cfg > dialplan.routing.emergency.x.* |
| Specify the server that will be used for routing calls..... | site.cfg > dialplan.routing.server.x.* |
| Configure the same parameters as above for a specific registration (overrides the global parameters above) | |
| | site.cfg > dialplan.x.* |
| Web Configuration Utility | |
| Specify impossible match behavior, trailing # behavior, digit map matching strings, and time-out value by navigating to Settings > SIP and expanding the Local Settings menu. | |

Understanding Digit Map Rules

The following is a list of digit map string rules. 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 will use to find the shortest possible match.

Digit map extension letter 'R' indicates that certain matched strings are replaced. Using a 'RRR' syntax, you can replace the digits between the first two 'R's with the digits between the last two 'R's. 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. The following examples illustrate the semantics of the syntax:

- **R9R604Rxxxxxxxx**—Replaces 9 with 604
- **xxR601R600Rxx**—When applied to 1160122 gives 1160022

- R9RRxxxxxxxx—Remove 9 at the beginning of the dialed number (replace 9 with *nothing*)
 - For example, if a customer dials 914539400, the first 9 is removed when the call is placed.
- RR604Rxxxxxxxx—Prepend 604 to all seven digit numbers (replace *nothing* with 604)
 - For example, if a customer dials 4539400, 604 is added to the front of the number, so a call to 6044539400 is placed.
- xR60xR600Rxxxxxxxx—Replace any 60x with 600 in the middle of the dialed number that matches
 - For example, if a customer dials 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.
- 0xxxS | 33xxH—All four digit numbers starting with a 0 are placed using the SIP protocol, whereas all four digit numbers starting with 33 are placed using the H.323 protocol.



Note: VVX Phones Do Not Match 'H'

Only VVX 1500 phones will match the 'H'. On all other phones, the 'H' is ignored and users will need to press the Send soft key to complete dialing. For example, if the digit map is '33xxH', the result is as follows:

- If a VVX 1500 user dials '3302' on an H.323 or dual protocol line, the call will be placed after the user dials the last digit.
- If a SoundPoint IP 650 user dials '3302', the user must press the **Send** soft key to complete dialing.

The following guidelines should be noted:

- 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 will be 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 2Rs or 5Rs) is considered an invalid digit map.

- If you use T in the left part of 'RRR' syntax, the digit map will not work. For example, R0TR322R will not work.

Microphone Mute

All phones have a microphone mute button. When you activate microphone mute, a red LED will glow or a mute icon will display on the phone screen, depending on the phone model you are using.

No configuration changes can be made to the microphone mute feature.

Using the Speed Dial Feature

You can link entries in your local contact directory to speed dial contacts on the phone. The speed dial feature enables you to place calls quickly using dedicated line keys or from a speed dial menu. To set up speed dial through the phone's contact directory, see [Using the Local Contact Directory](#). Speed dial configuration is also explained briefly in [Table 6-14: Using the Speed Dial Feature](#). In order to set up speed dial contacts, you will need to become familiar with [Table 6-12: Understanding the Local Contact Directory](#), which identifies the directory XML file and the parameters you need to set up your speed dial contacts.

The speed dial index range is 1 to 99 on SoundPoint IP 321, 331, 335 desktop phones and SoundStation IP 5000 and 6000 conference phones. For all other phones, the range is from 1 to 9999.

On some call servers, enabling Presence for an active speed dial contact will display that contact's status on the speed dial's line key label. For information on how to enable Presence for contacts, see [Using the Presence Feature](#).

Table 6-14: Using the Speed Dial Feature

Central Provisioning Server

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. Note that line keys are not available on the SoundStation IP 5000 or 6000 phones.

The template contact directory file**000000000000-directory~.xml**

Local Phone User Interface

New directory entries are assigned to the Speed Dial Index in numerical order. To assign a speed dial index to a contact, navigate go to the **Contact Directory**, highlight the contact, press the **Edit** soft key, and specify a **Speed Dial Index**.

**Power Tip: Quick Access to the Speed Dial List**

To quickly access the Speed Dial list, press the phone's Up arrow key from the idle display.

Example Speed Dial Configuration

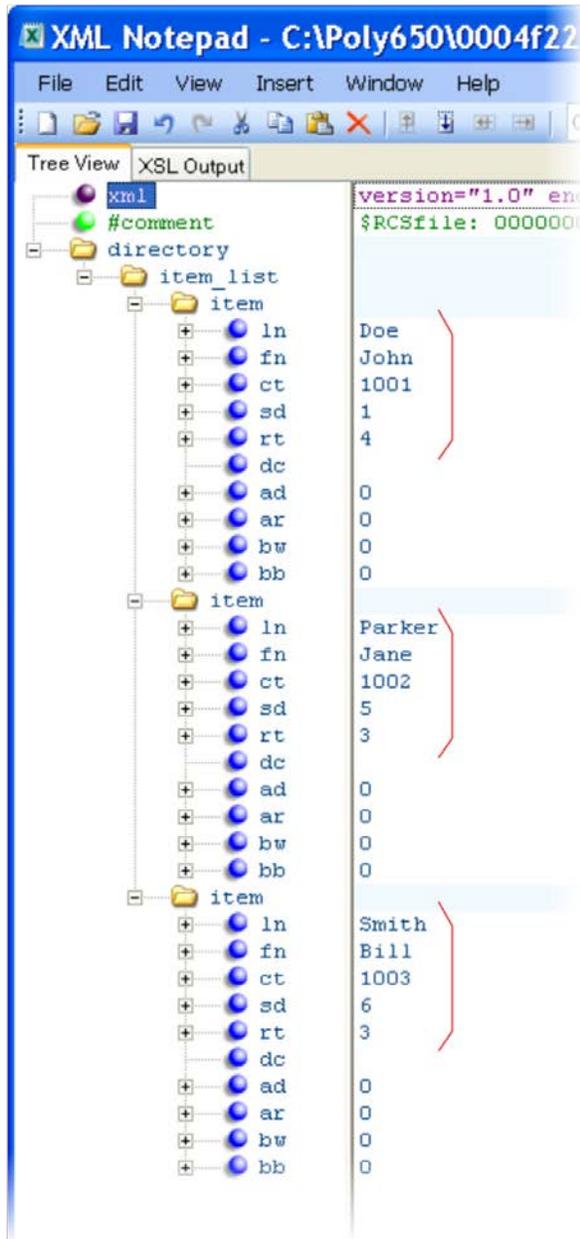
The first time you deploy and reboot the phones with UC Software, a template contact directory file named **00000000000-directory~.xml** is loaded to the provisioning server. You can edit and use this template file as a global contact directory for a group of phones or you can create your own per-phone directory file. To create a global directory, locate the **00000000000-directory~.xml** template in your UC Software files and remove the tilde (~) from the file name. When you reboot, the phone substitutes the global file with its own **<MACaddress>-directory.xml** which is uploaded to the server. If you want to create a per-phone directory, replace **<000000000000>** in the global file name with the phone's MAC address, for example, **<MACaddress>-directory.xml**.

On each subsequent reboot, the phone will look for its own **<MACaddress>-directory.xml** and then look for the global directory. Contact directories stored locally on the phone may or may not override the **<MACaddress>-directory.xml** on the server depending on your server configuration. The phone will always look for a local directory or **<MACaddress>-directory.xml** before looking for the global directory.

For more information on how to use the template directory file **00000000000-directory~.xml**, see [Using the Local Contact Directory](#).

Once you have renamed the directory file as a per-phone directory, 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.

The example local contact directory file shown net is saved with the phone's MAC address and shows the contact *John Doe* with extension number *1001* as speed dial entry '1' on the phone.



This configuration results in the following speed dial keys on the phone.



Setting the Time and Date Display

A clock and calendar are enabled by default. You can display the time and date for your time zone in several formats, or you can turn it off altogether. 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. You will have to synchronize the phone to the Simple Network Time Protocol (SNTP) time server. Until a successful SNTP response is received, the phone will continuously flash the time and date to indicate that they are not accurate.

The time and date display on phones in PSTN mode will be set by an incoming call with a supported Caller ID standard, or when the phone is connected to Ethernet and you enable the turn on the date and time display.

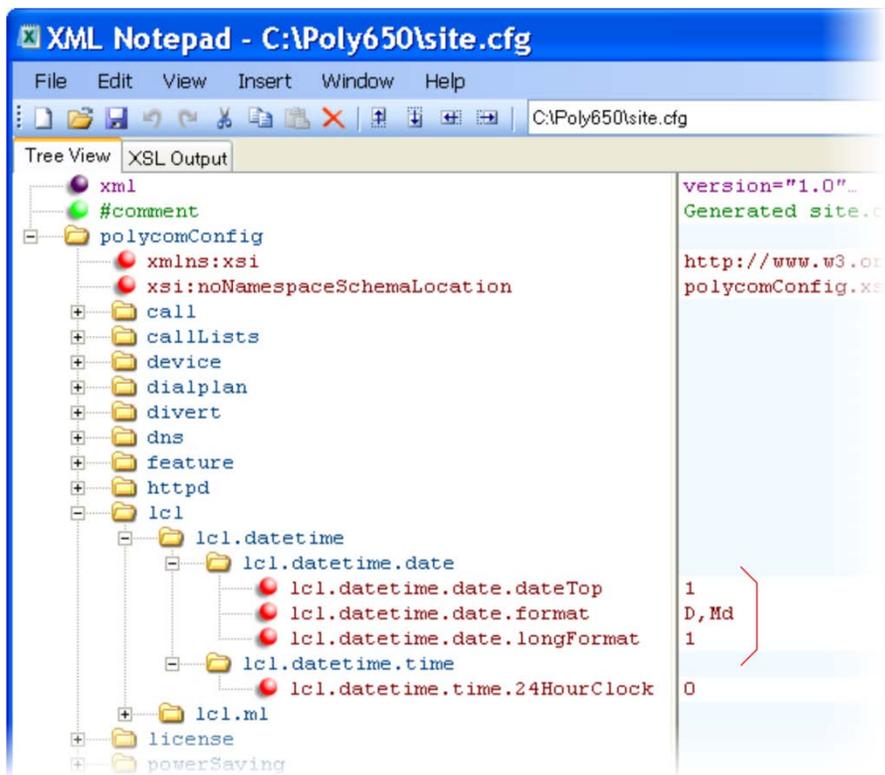
Note that the SoundPoint IP 321/331/335 and 450 phones have a limited selection of date formats due to their smaller screen size. See [Table 6-15: Setting the Time and Date Display](#) for basic time and display parameters.

Table 6-15: Setting the Time and Date Display

| Central Provisioning Server | template > parameter |
|---|--|
| Turn the time and date display on or off. | reg-advanced.cfg and site.cfg > up.localClockEnabled |
| Set the time and date display format. | site.cfg > lcl.datetime.date.* |
| Display time in the 24-hour format..... | site.cfg > lcl.datetime.time.24HourClock |
| Set the basic SNTP settings and daylight savings parameters. | site.cfg > tcplpApp.sntp.* |
| <hr/> | |
| Web Configuration Utility | |
| To set the basic SNTP and daylight savings settings navigate to Preferences > Date & Time . | |
| <hr/> | |
| Local Phone User Interface | |
| Basic SNTP settings can be made in the Network Configuration menu—see DHCP Menu or Network Interfaces Menu (Ethernet Menu) | |
| To set the time and date format and enable or disable the time and date display, press Menu > Settings > Basic > Preferences > Time & Date . | |

Example Configuration

The following shows an example configuration for the time and date display format. In this illustration, the date is set to display over the time and in long format. The 'D, Md' indicates the order of the date display, in this case, day of the week, month, and day. In this example, the default time format is used, or you can enable the 24 hour time display format.



This configuration results in the following time and date display format:



Use [Table 6-16: Date Formats](#) to choose values for the `lcl.datetime.date.format` and `lcl.datetime.date.longformat` parameters. The table shows values for Friday August 19th in 2011.

Table 6-16: Date Formats

| <code>lcl.datetime.date.format</code> | <code>lcl.datetime.date.longformat</code> | Date Displayed on Phone |
|---------------------------------------|---|-------------------------|
| dM,D | 0 | 19 Aug, Fri |
| dM,D | 1 | 19 August, Friday |

| <i>lcl.datetime.date.format</i> | <i>lcl.datetime.date.longformat</i> | <i>Date Displayed on Phone</i> |
|---------------------------------|-------------------------------------|--------------------------------|
| Md,D | 0 | Aug 19, Fri |
| Md,D | 1 | August 19, Friday |
| D,dM | 0 | Fri, 19 Aug |
| D,dM | 1 | Friday, August 19 |
| DD/MM/YY | n/a | 19/08/11 |
| DD/MM/YYYY | n/a | 19/08/2011 |
| MM/DD/YY | n/a | 08/19/11 |
| MM/DD/YYYY | n/a | 08/19/2011 |
| YY/MM/DD | n/a | 11/08/19 |
| YYYY/MM/DD | n/a | 2011/08/11 |

Adding an Idle Display Image

SoundPoint IP and SoundStation IP phones can display an idle image, such as a company logo, on the phone's display screen. Typically, this feature is used for images that you have created. You can use BMP or uncompressed JPG image file formats. Note that you may need to resize your images to fit the phone screen and that the idle display image will move the time and date display to the top of the phone screen. You can also create phone model-specific parameters to apply images to groups of phones; see the Example Idle Display Image Configuration for definitions of phone-specific parameters. Idle display images you apply with a phone model-specific parameter will override the `bitmap.idleDisplay.name` parameter which will be applied to other phones. If you want to add an image to your phone's idle display screen, see [Table 6-17: Adding an Idle Display Image](#) for parameters.

As of Polycom UC Software 3.3.0, you cannot use customized animations.

The VVX 500 and 1500 phones do not support the Idle Display Image feature.

Note that whereas an idle display image displays on a portion of the phone's screen, a graphic display background will display on the entire screen (see [Setting a Graphic Display Background](#)); line and soft key labels will display over the backgrounds.



Web Info: Adding an Idle Display Image

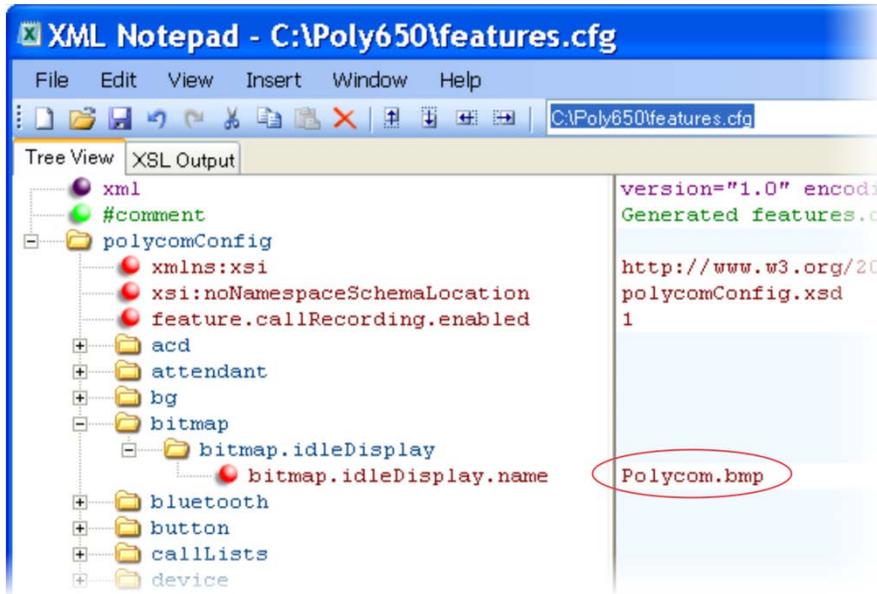
For detailed instructions on how to add a custom idle display logo to your phone, see [Adding a Custom Idle Display Logo to Polycom® SoundPoint® IP and SoundStation® IP Phones \(Technical Bulletin 18292\)](#).

Table 6-17: Adding an Idle Display Image

| | |
|---|--|
| Central Provisioning Server | template > parameter |
| Specify the file path of the idle display image | features.cfg > bitmap.idleDisplay.name |

Example Idle Display Image Configuration

The following illustration shows you how to display an idle image to your phone screen. In this example, in the **features.cfg** template, a Polycom logo is added in JPG image file format.



When you have added an idle image to your phone, the image will display on the phone screen, as shown next. Note that the image may not display exactly as shown; you may have to resize your image to fit the phone.



If you want to apply different idle display images to different phone models, you will need to create additional phone-specific parameters, shown next.

As shown in the following figure, the SoundPoint IP 650 will use `logo_main.bmp`, the SoundStation IP 6000 will use `logo_small.bmp`, and all other phones will use `logo_.bmp`.



For a list of all phone-specific parameter names, see [Product, Model, and Part Number Mapping](#).

Ethernet Switch

SoundPoint IP phones (except the SoundPoint IP 321) and the VVX 1500 phones have two Ethernet ports—labeled LAN and PC—and an embedded Ethernet switch that runs at full line rate. SoundStation IP phones and the SoundStructure VoIP Interface have one Ethernet port, labeled LAN. The Ethernet switch enables you to connect a personal computer and other Ethernet devices to the office LAN by daisy-chaining through the phone, eliminating the need for a stand-alone hub.

Each phone can be powered through an AC adapter or through a Power over Ethernet (PoE) cable connected to the phone's LAN port. To disable the PC Ethernet port, see [Disabling the PC Ethernet Port](#).

If you are not using VLAN and you have a device connected to a PC port, the SoundPoint IP switch gives higher transmit priority to packets originating in the phone. If you are using a VLAN, ensure that the 802.1p priorities for both default and real-time transport protocol (RTP) packet types are set to 2 or greater so that audio packets from the phone will have priority over packets from the PC port. For more information, see [<qos/>](#).

Setting a Graphic Display Background

You can display an image or a design on the background of the graphic display of all SoundPoint IP 450, 550, 560, and 650, VVX 500 and 1500 phones, and SpectraLink handsets. [Table 6-18: Setting a Graphic Display Background](#) links you to parameters and definitions in the reference section. Note that whereas an idle display image displays on a portion of the phone's screen (see [Adding an Idle Display Image](#)), a Graphic Display Background will display on the entire screen; the time and date and line and soft key labels will display over the backgrounds.



Note: Choosing a Graphic Display Background

Depending on the image you use, the graphic display background may affect the visibility of text and numbers on the phone screen. As a general rule, backgrounds should be light in shading for better phone and feature usability.

For SoundPoint IP 450, 550, 560, and 650 phones:

- You can choose from several default backgrounds. The phone supports BMP and JPEG file formats. The sizes of the LCD displays are listed in [Key Features of Your Polycom Phones](#) in Chapter 1. You can change the color backgrounds, import a picture of your choice, and you can modify the existing color and picture backgrounds. You can also modify the colors of the soft keys and line keys.

For VVX 500 and 1500 phones:

- The VVX phones display a default background picture. You can select your own background picture or design, or you can import a custom image. You can also select images from the Picture Frame (see [Configuring the Digital Picture Frame](#)).
- The VVX 1500 phones supports JPEG, BMP, and PNG image file formats up to a maximum size of 800x480 pixels. The phone may not correctly display larger images. Progressive/multiscan JPEG images are not supported.



Web Info: Adding a Graphic Display Background

For instructions on customizing the background on a SoundPoint IP phone, see [Technical Bulletin 62473: Customizing the Display Background on Your Polycom SoundPoint IP Phone](#).

For detailed instructions on adding a graphic display to a VVX phone, see [Technical Bulletin 62470: Customizing the Display Background on Your Polycom VVX 1500 Business Media Phone](#).

Table 6-18: Setting a Graphic Display Background

| | |
|---|-----------------------------------|
| Central Provisioning Server | template > parameter |
| Specify a background to display for your phone type | features.cfg > bg.* |
| Modify the color of the line and soft keys..... | features.cfg > button.* |
| Web Configuration Utility | |
| Specify which background to display by navigating to Preferences > Background | |
| Local Phone User Interface | |
| To select a background, on the phone, navigate to Menu > Settings > Basic > Preferences > Background > Select Background . | |
| On the VVX 500 and 1500, the user can save one of the Picture Frame images as the background by selecting Save as Background on the touch screen (see Configuring the Digital Picture Frame). | |
| To modify the color of the line and soft keys on your SoundPoint IP or SoundStation IP phones, navigate to Menu > Settings > Basic > Preferences > Label Color . | |

Example Graphic Display Background Configuration

This example configuration shows a background image applied to a SoundPoint 650 phone. The default background in the **features.cfg** template file, specified in the `bg.hiRes.gray.selection` parameter, is set to `2,1`. Where `2 = bg.hiRes.gray.pat.solid.*` and `1 = bg.hiRes.gray.pat.solid.1.*`, the phone will display the solid color specified by the RGB color pattern, in this case the color named *White*. In this example, the `bg.hiRes.gray.selection` parameter has been set to `3,6`. Where `3 = bg.hiRes.gray.bm.*` and `6 = bg.hiRes.gray.bm.6.*`, the phone will display the image named *Mountain.jpg*. In addition, the `bg.hiRes.gray.bm.6.adj` parameter has been changed to `-2` to lighten the background image so as not to conflict with the time and date display.

XML Notepad - C:\Poly650\features.cfg

File Edit View Insert Window Help

C:\Poly650\features.cfg

Tree View XSL Output

| | |
|---------------------------------|-------------------|
| xsi:noNamespaceSchemaLocation | polycomConfig.xsd |
| feature.callRecording.enabled | 1 |
| acd | |
| attendant | |
| bg | |
| bg.color | |
| bg.hiRes | |
| bg.hiRes.color | |
| bg.hiRes.gray | |
| bg.hiRes.gray.selection | 3, 6 |
| bg.hiRes.gray.bm | |
| bg.hiRes.gray.bm.1.adj | 0 |
| bg.hiRes.gray.bm.1.em.name | LeafEM.jpg |
| bg.hiRes.gray.bm.1.name | logo.bmp |
| bg.hiRes.gray.bm.2.adj | -3 |
| bg.hiRes.gray.bm.2.em.name | SailboatEM.jpg |
| bg.hiRes.gray.bm.2.name | Sailboat.jpg |
| bg.hiRes.gray.bm.3.adj | 0 |
| bg.hiRes.gray.bm.3.em.name | BeachEM.jpg |
| bg.hiRes.gray.bm.3.name | Beach.jpg |
| bg.hiRes.gray.bm.4.adj | -3 |
| bg.hiRes.gray.bm.4.em.name | PalmEM.jpg |
| bg.hiRes.gray.bm.4.name | Palm.jpg |
| bg.hiRes.gray.bm.5.adj | 2 |
| bg.hiRes.gray.bm.5.em.name | JellyfishEM.jpg |
| bg.hiRes.gray.bm.5.name | Jellyfish.jpg |
| bg.hiRes.gray.bm.6.adj | -2 |
| bg.hiRes.gray.bm.6.em.name | MountainEM.jpg |
| bg.hiRes.gray.bm.6.name | Mountain.jpg |
| bg.hiRes.gray.pat | |
| bg.hiRes.gray.pat.solid | |
| bg.hiRes.gray.pat.solid.1.blue | 255 |
| bg.hiRes.gray.pat.solid.1.green | 255 |
| bg.hiRes.gray.pat.solid.1.name | White |
| bg.hiRes.gray.pat.solid.1.red | 255 |
| bg.hiRes.gray.pat.solid.2.blue | 160 |
| bg.hiRes.gray.pat.solid.2.green | 160 |
| bg.hiRes.gray.pat.solid.2.name | Light Gray |
| bg.hiRes.gray.pat.solid.2.red | 160 |
| bg.hiRes.gray.pr | |
| bg.hiRes.gray.pr.1.adj | -3 |
| bg.hiRes.gray.pr.4.adj | 2 |
| bg.medRes | |
| bg.VVX_1500 | |
| bitmap | |
| blueooth | |

This example configuration will result in the following graphic display background on the phone screen. Note that line and soft key labels will display over the background image.



Enabling Multikey Answer

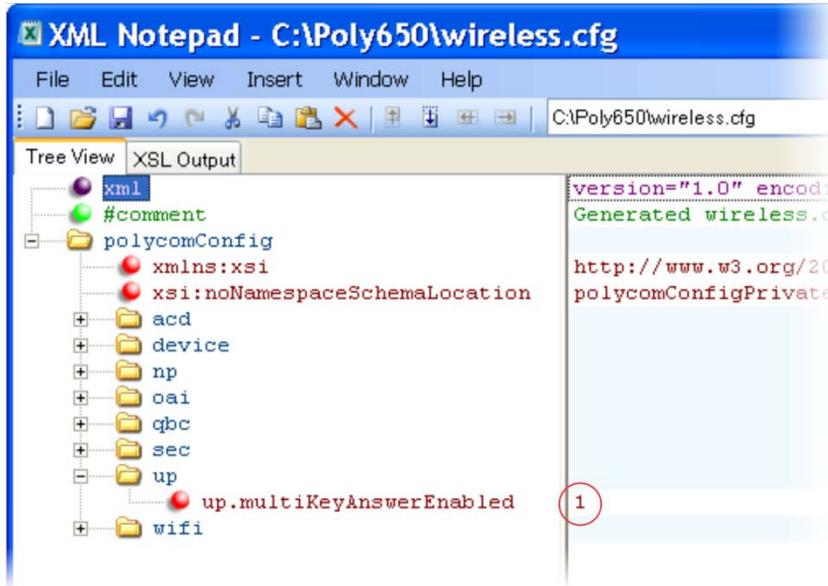
The SpectraLink handsets support the Multikey Answer feature. This feature enables you to answer incoming calls by pressing any key on the phone's keypad. [Table 6-19: Enabling Multikey Answer](#) links you to the parameter. You cannot use the Multikey Answer feature for Open Application Interface (OAI) calls, Group Paging, or Push-to-Talk (PTT) calls.

Table 6-19: Enabling Multikey Answer

| | |
|---|---|
| Central Provisioning Server | template > parameter |
| Enable or disable Multikey Answer | wireless.cfg > up.multiKeyAnswerEnabled |
| Web Configuration Utility | |
| To enable or disable Multikey Answer, navigate to Preferences > Additional Preferences and expand the User Preferences menu. | |

Example Multikey Answer Configuration

The following illustration shows you how to enable the Multikey Answer feature. The following configuration parameter is located in the **wireless.cfg** template.



Enabling Automatic Off-Hook Call Placement

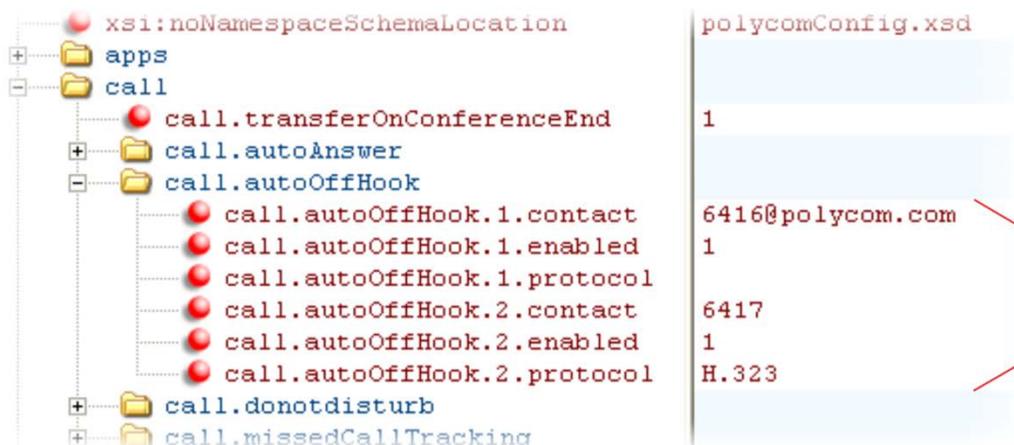
You can configure the phone to automatically place a call to a specified number when you go off-hook. This feature is sometimes referred to as *Hot Dialing*. The phone goes off-hook when you lift the handset, press the New Call soft key, or press the headset or speakerphone buttons on the phone. As shown in [Table 6-20: Enabling Automatic Off-Hook Call Placement](#), you can specify an off-hook call contact, and enable or disable the feature for specific line registrations. If you are using the VVX 1500, you can specify whether the automatic call uses the SIP (audio only) protocol or the H.323 (video) protocol.

Table 6-20: Enabling Automatic Off-Hook Call Placement

| Central Provisioning Server | template > parameter |
|---|---|
| Specify the contact to dial when the phone goes off-hook. . reg-advanced > | call.autoOffHook.x.contact |
| Enable or disable automatic off-hook call placement on registration x.
..... | reg-advanced > call.autoOffHook.x.enabled |
| Specify the call protocol for the VVX 1500 to use..... | reg-advanced > call.autoOffHook.x.protocol |

Example Automatic Off-Hook Placement Configuration

In the example configuration shown next, the automatic off-hook call placement feature has been enabled for registration 1 and registration 2. If registration 1 goes off-hook, a call will be automatically placed to `6416@polycom.com`, the contact that has been specified for registration 1 in `call.autoOffHook.1.contact`. Similarly, if registration 2 goes off-hook, a call will be automatically placed to `6417`. If the phone is a VVX 1500, registration 2 will automatically place a call using the H.323 protocol instead of the SIP protocol. Other phones will ignore the `protocol` parameter.



Enabling Call Hold

The purpose of call hold is to pause activity on one call so that you can use the phone for another task, for example, to place or receive another call or to search your phone's menu for information. See [Table 6-21: Enabling Call Hold](#) for a list of available parameters you can configure for this feature. When you place an active call on hold, a message will inform the held party that they are on hold. You can also configure a call hold alert to remind you after a period of time that a call is still on hold.

As of SIP 3.1, if supported by the call server, you can enter a music-on-hold URI. For more information, see draft RFC [Music on Hold draft-worley-service-example](#).

Table 6-21: Enabling Call Hold

| Central Provisioning Server | template > parameter |
|--|--|
| Specify whether to use RFC 2543 (c=0.0.0.0) or RFC 3264 (a=sendonly or a=inactive) for outgoing hold signaling | <code>sip-interop.cfg > volpProt.SIP.useRFC2543hold</code> |
| Specify whether to use sendonly hold signaling | <code>sip-interop.cfg > volpProt.SIP.useSendonlyHold</code> |
| Configure local call hold reminder options | <code>sip-interop.cfg > call.hold.localReminder.*</code> |
| Specify the music-on-hold URI | <code>sip-interop.cfg > volpProt.SIP.musicOnHold.uri</code> |

Local Phone User Interface

Navigate to **Menu > Settings > Advanced > Administration Settings > SIP Server Configuration (Call Server Configuration > SIP** on the VVX 1500 phone) to specify whether or not to use RFC 2543 (c=0.0.0.0) outgoing hold signaling. The alternative is RFC 3264 (a=sendonly or a=inactive).

Example Call Hold Configuration

The following two illustrations show a sample configuration for the call hold feature. Both illustrations are taken from the **sip-interop.cfg** template. In the first illustration, the three `localReminder.*` parameters have been configured to play a tone to remind you of a party on hold, that the tone will begin to play 45 seconds after you put a party on hold, and that the tone will repeat every 30 seconds.

| Parameter | Value |
|---|-------|
| <code>call.dialtoneTimeout</code> | 60 |
| <code>call.directedCallPickupMethod</code> | |
| <code>call.directedCallPickupString</code> | *97 |
| <code>call.enableOnNotRegistered</code> | 1 |
| <code>call.lastCallReturnString</code> | *69 |
| <code>call.localConferenceCallHold</code> | 0 |
| <code>call.localConferenceEnabled</code> | 1 |
| <code>call.offeringTimeout</code> | 60 |
| <code>call.parkedCallRetrieveMethod</code> | |
| <code>call.parkedCallRetrieveString</code> | |
| <code>call.rejectBusyOnDnd</code> | 1 |
| <code>call.ringBackTimeout</code> | 60 |
| <code>call.singleKeyPressConference</code> | 0 |
| <code>call.stickyAutoLineSeize</code> | 0 |
| <code>call.urlModeDialing</code> | 0 |
| <code>call.advancedMissedCalls</code> | |
| <code>call.autoRouting</code> | |
| <code>call.callWaiting</code> | |
| <code>call.clickToDial</code> | |
| <code>call.hold</code> | |
| <code>call.hold.localReminder</code> | |
| <code>call.hold.localReminder.enabled</code> | 1 |
| <code>call.hold.localReminder.period</code> | 30 |
| <code>call.hold.localReminder.startDelay</code> | 45 |
| <code>call.hold.remoteNotification</code> | |
| <code>call.shared</code> | |

In the second illustration, the `musicOnHold.uri` parameter has been configured so the party on hold will hear music played from SIP URI `moh@example.com`.



Using Call Transfer

The Call Transfer feature enables you to transfer an existing active call to a third-party address using a Transfer soft key. For example, if party A is in an active call with party B, party A can

transfer party B to party C (the third party). In this case, party B and party C will begin a new call and party A will disconnect. [Table 6-22: Using Call Transfer](#) shows you how to specify call transfer behavior.

You can perform two types of call transfers:

- **Blind Transfer** Party A transfers the call without speaking to party C.
- **Consultative Transfer** Party A speaks to party C before party A transfers the call.

By default, a Transfer soft key will display when party A calls Party C and Party C’s phone is ringing, the proceeding state. In this case, party A has the option to complete the transfer before party C answers, which ends party A’s connection to party B and C. You can disable this option so that the Transfer soft key does not display during the proceeding state. In this case, party A can either wait until party C answers or press the Cancel soft key and return to the original call.

Table 6-22: Using Call Transfer

| Central Provisioning Server | template > parameter |
|--|----------------------|
| Specify whether to allow transfers while calls are in a proceeding state
..... sip-interop.cfg > volpProt.SIP.allowTransferOnProceeding | |
| Specify whether the default transfer type is blind or consultative (SoundPoint IP 321/331/335 only)
..... sip-interop.cfg > call.transfer.blindPreferred | |

Example Call Transfer Configuration

In the following example configuration, the parameter `allowTransferOnProceeding` has been disabled so that the Transfer soft key will not display while the third-party phone is ringing, the proceeding state. Once you have connected to the third-party, the Transfer soft key will display. If the third-party does not answer, you can press the Cancel soft key to return to the active call.



Creating Local and Centralized Conferences

You can set up local or centralized conferences. Local conferences require a host phone, which processes the audio of all parties. All phones support three-party local conferencing. Alternatively, you can use an external audio bridge, available via a central server, to create a centralized conference call. Polycom recommends using centralized conferencing to host four-party conferences, though some phones do enable to host four-party conferences locally. The SoundPoint IP 450, 550, 560, and 650 phones support four-way local conferencing. The SoundPoint IP 321, 331, and 335, SoundStation IP phones, and VVX 500 phones support three-way local conferencing.

See the parameters in [Table 6-23: Creating Local and Centralized Conferences](#) to set up a conference type and the options available for each type of conference. 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 will perform 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 [Enabling Conference Management](#).

Table 6-23: Creating Local and Centralized Conferences

| Central Provisioning Server | template > parameter |
|---|--|
| Specify whether, during a conference call, the host can place all parties or only the host on hold
..... | sip-interop.cfg > call.localConferenceCallHold |
| Specify whether or not the remaining parties can communicate after the conference host exits the conference | sip-interop.cfg > call.transferOnConferenceEnd |
| Specify whether or not all parties hear sound effects while setting up a conference
..... | sip-interop.cfg > call.singleKeyPressConference |
| Specify which type of conference to establish and the address of the centralized conference resource
..... | sip-interop.cfg > volpProt.SIP.conference.address |

Enabling Conference Management

This feature enables you to add, hold, mute, and remove conference participants, as well as obtain additional information about participants. Use the parameters listed in [Table 6-24: Managing Conferences](#) to configure how you want to manage conferences. VVX 1500 users can choose which conference call participants to exchange video with. If you are using the SoundStation Duo in PSTN mode, you can set up a conference but the conference management feature is not available.

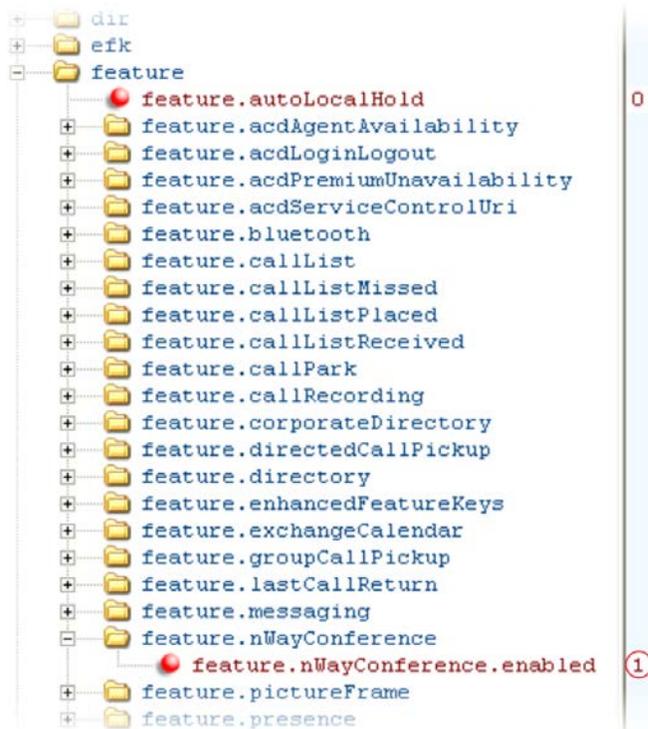
This feature is supported on the SoundPoint IP 450, 550, 560, and 650, the VVX 500 and 1500, and the SoundStation Duo in SIP mode.

Table 6-24: Managing Conferences

| Central Provisioning Server | template > parameter |
|--|---|
| Enable or disable the conference management feature. | |
| | features.cfg > feature.nWayConference.enabled |

Example Conference Management Configuration

The following example shows you how to enable the conference management feature in the **features.cfg** file.



When you enable conference management, a **Manage** soft key will display on the phone during a conference. When you press the **Manage** soft key, the **Manage Conference** screen, shown next, will display with soft keys you can use to manage conference participants.



Configuring Call Forwarding

The phone provides a flexible call forwarding feature that enables you to forward incoming calls to another destination. You can apply call forwarding in the following ways:

- To all calls
- To incoming calls from a specific caller or extension
- When your phone is busy
- When Do Not Disturb is enabled
- When the phone has been ringing for a specific period of time
- You can have incoming calls forwarded automatically to a predefined destination you choose or you can manually forward calls to a destination.

You will find parameters for all of these options in [Table 6-25: Configuring Call Forwarding](#).

To enable server-based call forwarding, you must enable the feature on both a registered phone and on the server and the phone is registered. If you enable server-based call forwarding on one registration, other registrations will not be affected. Server-based call forwarding disables local Call Forward and DND features.

Server-based call forwarding will behave the same as pre-SIP 2.1 feature with the following exception:

- If server-based call forwarding is enabled, but inactive, and you press the call forward soft key, the 'moving arrow' icon will not display on your phone and incoming calls will not be forwarded.



Troubleshooting: Call Forwarding Does Not Work on My Phone

The server-based and local call forwarding features do not work with the Shared Call Appearance (SCA) and Bridged Line Appearance (BLA) features. If you have SCA or BLA enabled on your phone, you will need to disable the feature before you can use call forwarding.

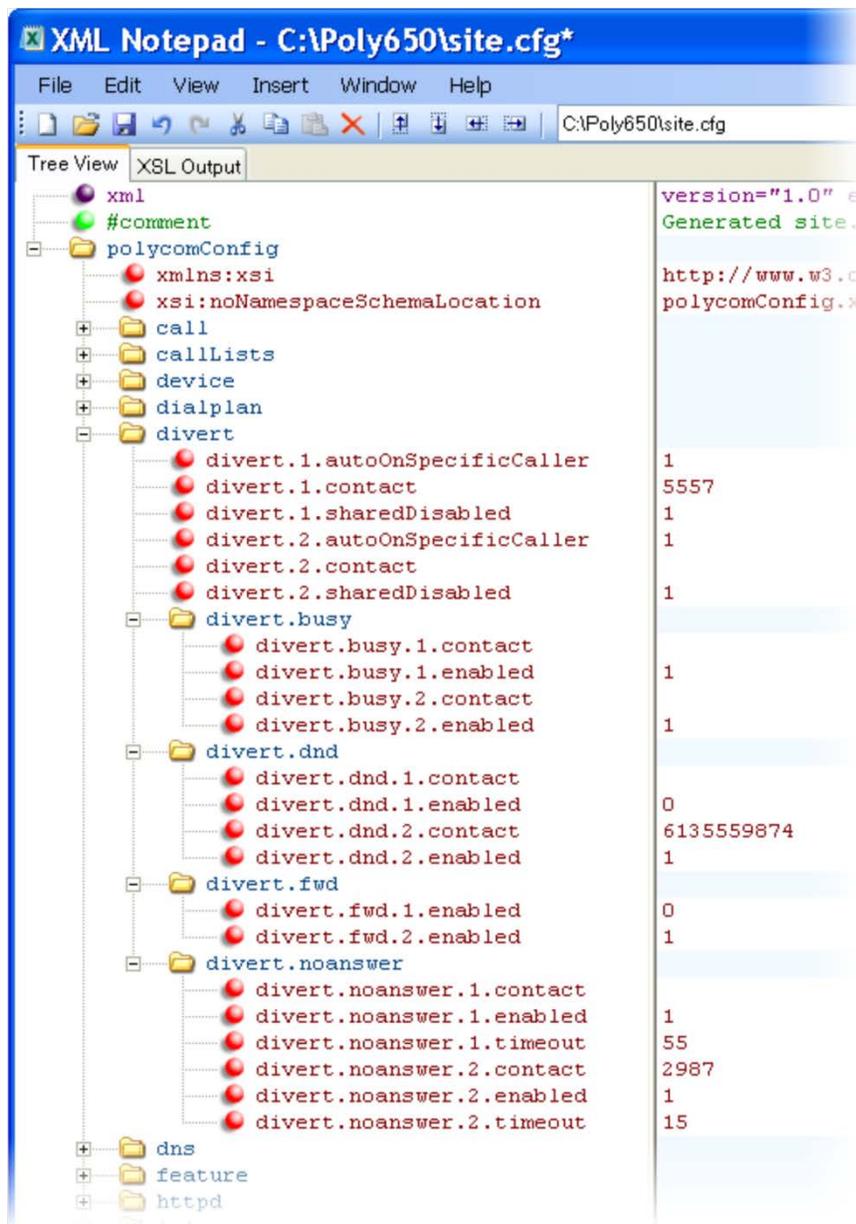
The call server uses the Diversion field with a SIP header to inform the phone of a call's history. For example, when you enable call forwarding, the Diversion header allows the receiving phone to indicate who the call was from, and the phone number it was forwarded from.

Table 6-25: Configuring Call Forwarding

| Central Provisioning Server | template > parameter |
|--|--|
| Enable or disable server-based call forwarding | sip-interop.cfg > volpProt.SIP.serverFeatureControl.cf |
| Enable or disable local call forwarding behavior when server-based call forwarding is enabled | sip-interop.cfg > volpProt.SIP.serverFeatureControl.localProcessing.cf |
| Enable or disable the display of the Diversion header and the order in which to display the caller ID and number | sip-interop.cfg > volpProt.SIP.header.diversion.* |
| Set all call diversion settings including a global forward-to contact and individual settings for call forward all, call forward busy, call forward no-answer, and call forward do-not-disturb | site.cfg > divert.* |
| Enable or disable server-based call forwarding as a per-registration feature | reg-advanced.cfg > reg.x.fwd.* |
| Web Configuration Utility | |
| To set all call diversion settings navigate to Settings > Lines , select a line from the left pane, and expand the Call Diversion menu. | |
| Local Phone User Interface | |
| To enable and set call forwarding from the phone, navigate to Menu > Features > Forward . | |

Example Call Forwarding Configuration

In the example configuration shown next, the call forwarding parameters for registration 1 have been changed from the default values. The forward-always contact for registration 1 is 5557 and this number will be used if the parameters `divert.busy`, `divert.dnd`, or `divert.noanswer` are not set. Parameters you set in those fields will override `divert.1.contact`. To enable these three divert options for each registration, you will need to enable the `divert.fwd.x.enabled` parameter and the `.enabled` parameter for each of the three forwarding options you want to enable. In this example, `divert.fwd.1.enabled` has been disabled; all calls to registration 1 will be diverted to 5557 and you do not have the option of enabling any of the three forwarding options on the phone. The three divert options are enabled for registration 2 in the `divert.fwd.2.enabled` parameter, giving you the option to enable or disable any one of the three forwarding options on the phone. When do not disturb (DND) is turned on, you can set calls to registration 2 to be diverted to 6135559874 instead of 5557. The parameter `divert.noanswer.2.enabled` is enabled so that, on the phone, you can set calls to registration 2 that ring for more than 15 seconds, specified in `divert.noanswer.2.timeout`, to be diverted to 2987, as set in `divert.noanswer.2.contact`.



Configuring Directed Call Pick-Up

This feature enables you 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. [Table 6-26: Configuring Directed Call Pickup](#) lists the configuration parameters for the directed call pick-up feature.

Table 6-26: Configuring Directed Call Pickup

| Central Provisioning Server | template > parameter |
|---|---|
| Turn this feature on or off..... | features.cfg > feature.directedCallPickup.enabled |
| Specify the type of directed call pick-up..... | sip-interop.cfg > call.directedCallPickupMethod |
| Specify the star code to initiate a directed call pickup | sip-interop.cfg > call.directedCallPickupString |
| Determine the type of SIP header to include | sip-interop.cfg > volpProt.SIP.strictReplacesHeader |

Example Directed Call Pickup Configuration

The configuration parameters for the directed call pickup feature are located in two template files. You enable directed call pickup in the **features.cfg** template file and configure the feature using the **sip-interop.cfg** file.

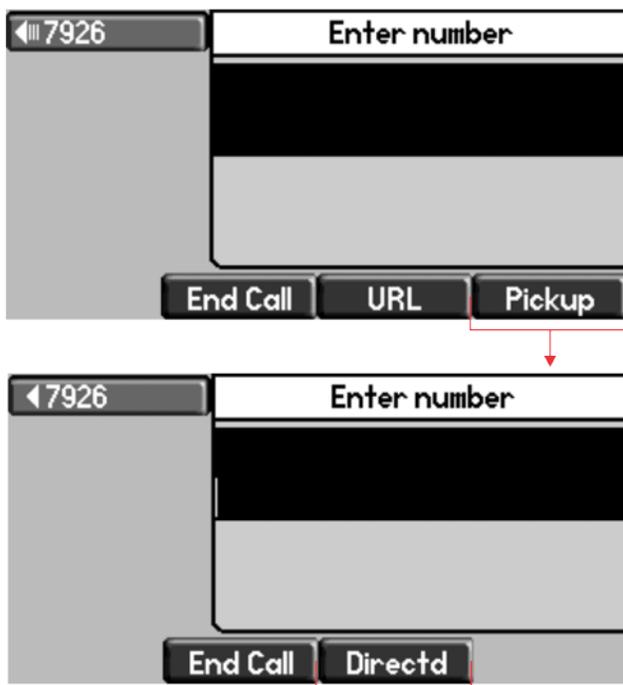
In the following configuration example, the directed call pickup feature has been enabled in the **features.cfg** template file:



Once directed call pickup is enabled, you can configure the feature using parameters located in the **sip-interop.cfg** template file. In the following illustration, the pickup method has been set to *native*, which means that the server is used for directed call pickup instead of the *PickupString*. If the pickup method was set to *legacy*, the pickup string **97* would be used by default. The pickup string can be different for different call servers, check with your call server provider if you configure legacy mode directed call pickup.

| | |
|---|---|
| <pre> xsi:noNamespaceSchemaLocation call call.dialtoneTimeout call.directedCallPickupMethod call.directedCallPickupString call.enableOnNotRegistered call.lastCallReturnString </pre> | <pre> polycomC 60 native *97 1 *69 </pre> |
|---|---|

When you enable directed call pickup, the phone will display a **Pickup** soft key when you go off-hook. When you press the **Pickup** soft key, the **Direct'd** soft key will display, as shown next.



Enabling Group Call Pickup

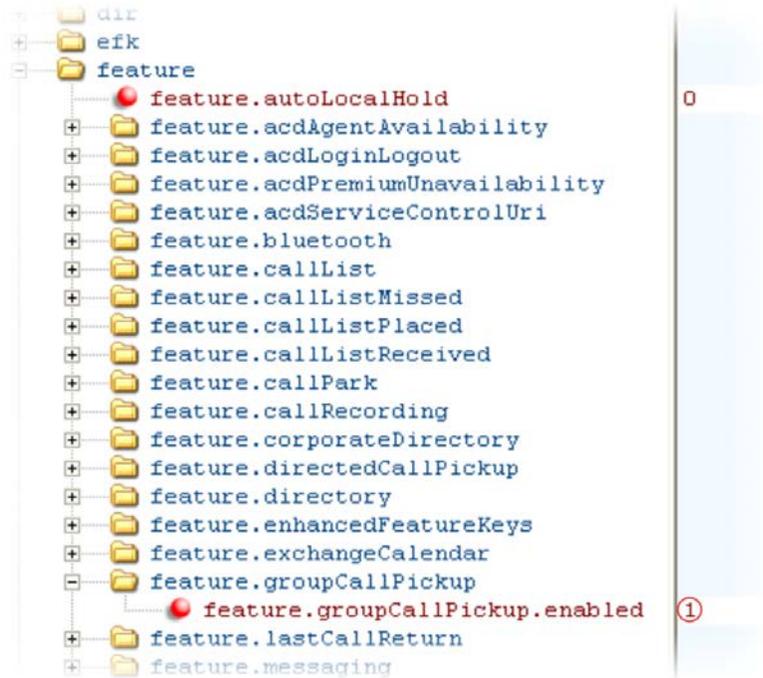
This feature enables you to pick up incoming calls to any phone within a predefined group of phones, without dialing the extension of another phone. The parameter to enable this feature is shown in [Table 6-27: Enabling Group Call Pickup](#). This feature requires support from a SIP server and setup 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.

Table 6-27: Enabling Group Call Pickup

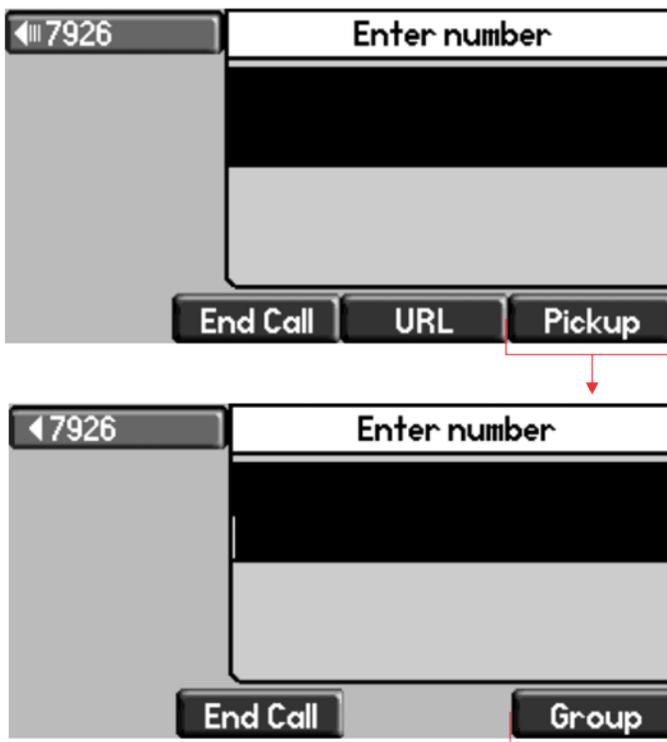
| | |
|------------------------------------|--|
| Central Provisioning Server | template > parameter |
| Turn this feature on or off..... | features.cfg > feature.groupCallPickup.enabled |

Example Group Call Pickup Configuration

The following illustration shows you how to enable the group call pickup feature in the `features.cfg` template.



When you enable the group call pickup, the phone will display a **Pickup** soft key when you go off-hook. If you select **Pickup**, the **Group** soft key is displayed.



Configuring Call Park and Retrieve

You can park an active call and retrieve parked calls from 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. See [Table 6-28: Configuring Call Park and Retrieve](#) for parameters you can configure.

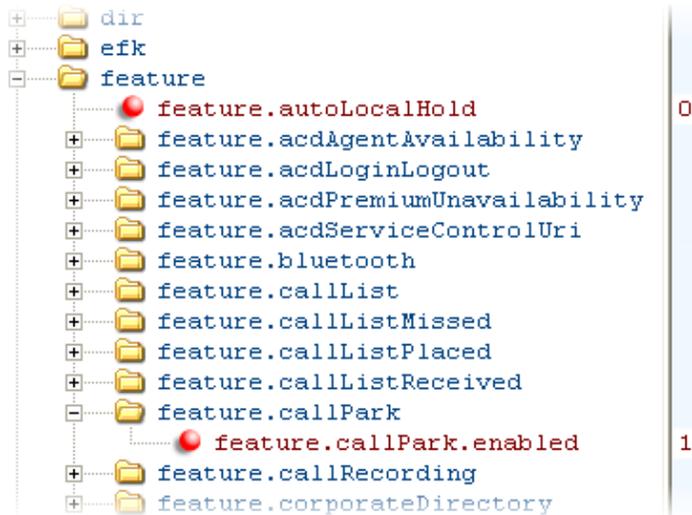
Table 6-28: Configuring Call Park and Retrieve

| Central Provisioning Server | template > parameter |
|--|---|
| Enable or disable call park and retrieve | <code>features.cfg > feature.callPark.enabled</code> |
| Specify the method the phone will use to retrieve a BLF call | <code>sip-interop.cfg > call.parkedCallRetrieveMethod</code> |
| Specify the star code used to retrieve a parked call | <code>sip-interop.cfg > call.parkedCallRetrieveString</code> |

Example Call Park and Retrieve Configuration

The configuration parameters for the call park and retrieve feature are located in two template files. You can enable the feature using the **features.cfg** template file and configure the feature using the **sip-interop.cfg** file.

In the following configuration example, the call park feature has been enabled in the **features.cfg** template file.



You can configure the call park and call retrieve feature using parameters located in the **sip-interop.cfg** template file. The following illustration shows that the parked call retrieve method has been set to `native`, meaning that the phone will use SIP INVITE with the Replaces header. The method can also be set to `legacy`, meaning that the phone will use the `call.parkedCallRetrieveString` star code to retrieve the parked call.



When the call park and retrieve feature is enabled, the Park soft key will display when you are in a connected call. To park the call, press the Park soft key and enter the number of the call orbit and park the call. To retrieve a parked call, go off-hook and press the Pickup soft key. Enter the number of the call orbit and press the Retrieve soft key, shown next.



Enabling Last Call Return

The phone supports redialing of the last received call. [Table 6-29: Enabling Last Call Return](#) shows you the parameters to enable this feature. 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.

Table 6-29: Enabling Last Call Return

| Central Provisioning Server | template > parameter |
|--|---|
| Enable or disable last call return..... | features.cfg > feature.lastCallReturn.enabled |
| Specify the string sent to the server for last-call-return | sip-interop.cfg > call.lastCallReturnString |

Example Configuration for Last Call Return

The configuration parameters for last call return feature are located in two template files. You can enable the feature using the **features.cfg** template file and configure the feature using the **sip-interop.cfg** file.

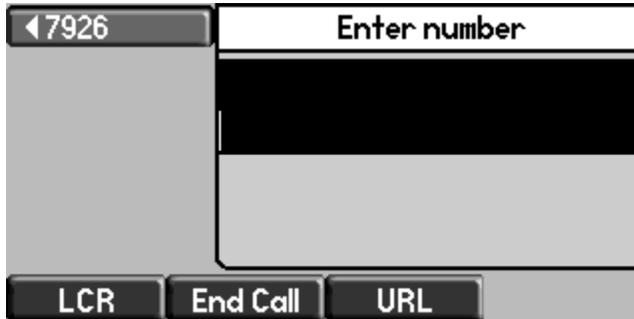
In the following configuration example, the last call return feature has been enabled in the **features.cfg** template file:

| | | |
|---|----------------------------------|---|
| + | efk | |
| - | feature | |
| | feature.autoLocalHold | 0 |
| + | feature.acdAgentAvailability | |
| + | feature.acdLoginLogout | |
| + | feature.acdPremiumUnavailability | |
| + | feature.acdServiceControlUri | |
| + | feature.bluetooth | |
| + | feature.callList | |
| + | feature.callListMissed | |
| + | feature.callListPlaced | |
| + | feature.callListReceived | |
| + | feature.callPark | |
| + | feature.callRecording | |
| + | feature.corporateDirectory | |
| + | feature.directedCallPickup | |
| + | feature.directory | |
| + | feature.enhancedFeatureKeys | |
| + | feature.exchangeCalendar | |
| + | feature.groupCallPickup | |
| - | feature.lastCallReturn | |
| | feature.lastCallReturn.enabled | 1 |
| + | feature.messaging | |

Once last call return is enabled, you can configure the feature using parameters located in the **sip-interop.cfg** template file. The following shows the default value for the `call.lastCallReturnString` parameter. The last call return string value depends on the call server you use. Consult with your call server provider for the last call return string.

| | | |
|---|-------------------------------|--------|
| | xsi:noNamespaceSchemaLocation | polyco |
| - | call | |
| | call.dialtoneTimeOut | 60 |
| | call.directedCallPickupMethod | |
| | call.directedCallPickupString | *97 |
| | call.enableOnNotRegistered | 1 |
| | call.lastCallReturnString | *69 |
| | call.localConferenceCallHold | 0 |
| | call.localConferenceEnabled | 1 |

When you enable the last call return feature, the phone will display an **LCR** soft key when it goes off-hook, as shown next. When you press the **LCR** soft key, you will place a call to the phone address that last called you.



When you select **Last Call Return**, you will place a call to the phone address that last called you.



Chapter 7: Setting Up Advanced Phone Features

After you set up your Polycom® phones with a default configuration on the network, phone users will be able to place and receive calls; however, you may want to make some changes to optimize your configuration for your organization and user's needs. Polycom provides basic and advanced features that you can configure for the phones. This chapter will show you how to configure all available advanced phone features, call server features, and Polycom and third-party applications.

Before you begin configuring phone features, take the time to read the short introductory section [Reading the Feature Parameter Tables](#). This section provides important information you need to know in order to successfully perform configuration changes.

This chapter shows you how to make configuration changes for the following advanced features:

- [Configuring the Phone's Keypad Interface](#) Enables you to change key functions from the factory defaults.
- [Assigning Multiple Line Keys Per Registration](#) You can assign multiple Line Keys to a single registration.
- [Enabling Multiple Call Appearances](#) All phones support multiple concurrent calls. You can place any active call on hold to switch to another call.
- [Customizing and Downloading Fonts](#) Enables you to customize the fonts used on the phone's display screen and download new fonts for your phone.
- [Setting the Phone Language](#) All phones have multilingual user interfaces.
- [Enabling Instant Messaging](#) Supports the sending and receiving of instant text messages.
- [Synthesized Call Progress Tones](#) Match the phone's call progress tones to a region.
- [Using the Microbrowser and Web Browser](#) SoundPoint IP 321, 331, 335, 450, 550, 560, and 650 desktop phones, SoundStation IP 5000 and 6000 conference phones, and VVX 500 and 1500 phones (pre-SIP 3.2.2) support an XHTML browser. The VVX 1500 phones running SIP 3.2.2 or later support a Webkit browser.
- [Configuring Real-Time Transport Protocol Ports](#) Phone treat all real time transport protocol (RTP) streams as bi-directional from a control perspective, and expect that both RTP end points will negotiate the respective destination IP addresses and ports.
- [Configuring Network Address Translation](#) Phones can work with certain types of network address translation (NAT).

- **Using the Corporate Directory** You can configure the phone to access your corporate directory if it has a standard LDAP interface. This feature is part of the Productivity Suite. Active Directory, OpenLDAP, Microsoft ADAM, and SunLDAP are currently supported.
- **CMA Directory** Enables you to access a corporate contact directory stored on the CMA server.
- **Recording and Playing Audio Calls** Enables you to record and play back any active conversation to a USB device. The files have a date and time stamped for easy archiving and can be played back on the phone or on any computer with a media playback program that supports the .wav format. This feature is part of the Productivity Suite.
- **Configuring the Digital Picture Frame** Display a slide show of images on the phone's idle screen.
- **Configuring Enhanced Feature Keys** Enables you to redefine soft keys to suit your needs. In SIP 3.0, this feature required a license key. In later releases, no license key is required.
- **Configuring Soft Keys** Enables you to create your own soft keys, and display them with or without the standard soft keys.
- **Enabling the Power Saving Feature** Enable and set hours for the power-saving feature.
- **Configuring Push-to-Talk and Group Paging** Send one-way page broadcasts or send and receive push-to-talk messages.
- **Flexible Line Key Assignment** Enables you to define any line key function to any line key location on the phone screen for the SoundPoint IP 450, 550, 560, and 650 phones.
- **Enabling Bridged Line Appearance** Allows a line extension or phone number to appear on multiple users' phones. This feature requires call server support.
- **Using Busy Lamp Field** You can monitor the hook status of remote parties with the busy lamp field (BLF) LEDs and you can display your status on an attendant console phone. This feature may require call server support.

The BLF feature was enhanced in SIP 3.2 as follows:

- To provide individual subscription-based BLF monitoring (without requiring the call server to maintain a centralized resource list).
- To allow the single button 'remote pick-up' feature to be implemented using Directed Call Pick-Up and SIP signaling, as well as the star code method supported in SIP 3.1.
- **Enabling Voicemail Integration** Enables access to compatible voice mail servers.
- **Enabling Multiple Registrations** SoundPoint IP desktop phones and VVX 500 and 1500 phones support multiple registrations per phone. However, SoundStation IP conference phones support a single registration.
- **Configuring SIP-B Automatic Call Distribution** Supports ACD agent available and unavailable and allows ACD login and logout. This feature requires call server support.

- **Configuring Feature Synchronized Automatic Call Distribution** Supports ACD agent availability and unavailability and allows ACD sign in and sign out. This feature requires call server support.
- **Setting Up Server Redundancy** Phones support server redundancy to ensure the continuity of phone service when the call server is offline for maintenance, fails, or the connection between the phone and server fails.
- **DNS SIP Server Name Resolution** Enter the DNS name for a proxy/registrar address.
- **Using the Presence Feature** Enables you to monitor the status of other users/devices, and for other users/devices to monitor you. This feature requires call server support.
- **Using CMA Presence** Monitor the status of other remote users and phones on the CMA directory.
- **Enabling Access URL in SIP Messages** Phones can receive a URL inside a SIP message (for example, as a SIP header extension in a SIP INVITE) and subsequently access the provided URL in the Web Browser.
- **Configuring the Static DNS Cache** Set up a cache for DNS information and provide for negative caching.
- **Displaying SIP Header Warnings** Displays a 'pop-up' warning message to the users from a SIP header message.
- **Quick Setup of Polycom Phones** Provides a simplified interface to enter provisioning server parameters while your phone boots.
- **Provisional Polling of Polycom Phones** Phones can be set to automatically check for software downloads using a random schedule or through a predefined schedule.

This chapter also shows you how to make configuration changes to support the following Polycom and third-party applications:

- **Setting Up Microsoft Office Communications Server 2007 R2 Integration** Use the Microsoft Office Communications Server (LCS) 2007 R2 to share ideas and information immediately with business contacts.
- **Setting Up Microsoft Lync Server 2010 Integration** You can use certain SoundPoint IP, SoundStation IP, and VVX phones with Microsoft Lync Server 2010 to immediately share ideas and information with business contacts. This feature requires call server support.
- **Enabling Polycom Desktop Connector Integration** Use your mouse and keyboard to enter information and navigate screens on your VVX 500 and 1500 phone running Polycom UC Software 4.0.1 or later.
- **Enabling Microsoft Exchange Calendar Integration** Enables users to manage meetings and reminders with your phone, and enables you to dial in to conference calls. This feature is supported only on VVX 500 and 1500 phones and SpectraLink handsets, and requires Microsoft Exchange Calendar Integration.

- [Configuring the Polycom Quick Barcode Connector](#) Captures and decodes barcode patterns with the SpectraLink handsets and transfer the data to applications running on one or more host computers.
- [Configuring the Open Application Interface](#) SpectraLink handsets can retrieve and respond to information on third-party computer applications.
- [Enabling Location Services](#) Use location services to send reports for Ekahau® Real-Time Location Systems (RTLS) on the SpectraLink handsets.

To troubleshoot any problems with your Polycom phones on the network, see [Troubleshooting Your Polycom Phones](#). For more information on the configuration files, see [Polycom UC Software Configuration Files](#). For more information on the Web Configuration Utility, see [Getting Started with the Polycom Web Configuration Utility](#). For instructions on how to read the feature descriptions in this section, see [Reading the Feature Parameter Tables](#).

Configuring the Phone's Keypad Interface

You can customize many of the default key functions on the phone's keypad interface. [Table 7-1: Configuring Phone Keys](#) lists the parameters you can configure to change the layout of your phone's keypad. Polycom recommends that you configure only those phone keys with removable key caps, including: Directories, Applications, Conference, Transfer, Redial, Menu, Messages, Do Not Disturb, and Call Lists.



Caution: Choosing Keys to Remap

Polycom recommends that you remap only those keys with removable key caps. If you remap other keys, your phone may not work properly. You should not remap the following keys: the dial pad, volume control, handsfree, mute, headset, hold, and the navigation arrow keys.

You can configure phone keys in the following ways:

- You can assign function or features to a key.
- You can turn a phone key into a speed dial.
- You can assign enhanced feature key (EFK) operations to a phone key. For example, you can reach a phone menu path to a single key press using a macro code. To find out how to configure EFK functions, see [Configuring Enhanced Feature Keys](#).
- You can delete all functions and features from a phone key.



Note: You Cannot Remap All Keys

The SpectraLink handsets have no removable key caps and you cannot customize the phone's keypad. Since there is no Redial key on the SoundPoint IP 321, 331, or 335 phones, the redial function cannot be remapped. SoundStation IP 5000 and 6000 keys cannot be remapped to behave as Speed Dial keys.

Table 7-1: Configuring Phone Keys

| Central Provisioning Server | template > parameter |
|---|--|
| Set the primary key function for key y on phone model x | features.cfg > key.x.function.prim |
| Set the secondary key function for key y on phone model x | features.cfg > key.x.subPoint.prim |

For an illustration of the default phone key configuration layout, see [Default Feature Key Layouts](#).

Assigning Multiple Line Keys Per Registration

You can assign a single registered phone line address to multiple line keys on SoundPoint IP, VVX phones, and SpectraLink handsets. See [Table 7-2: Multiple Line Keys Per Registration](#) for the parameter you need to set. This feature can be useful for managing a high volume of calls to a line. This feature is one of several features associated with *Flexible Call Appearances*. For the maximum number of line keys per registration for each phone model, and for definitions of all features associated with Flexible Call Appearances, see [Table 7-4: Flexible Call Appearances](#).

Table 7-2: Multiple Line Keys Per Registration

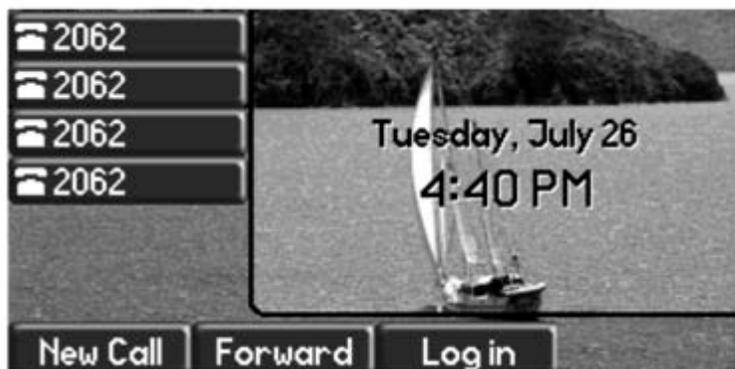
| Central Provisioning Server | template > parameter |
|--|---|
| Specify the number of line keys to use for a single registration | reg-advanced.cfg > reg.x.lineKeys |
| Web Configuration Utility | |
| To assign the number of line keys per registration, navigate Settings > Lines , select the number of lines from the left pane, expand Identification , and edit Number of Line Keys . | |
| Local Phone User Interface | |
| Assign the number of line keys per registration by navigating to Menu > Settings > Advanced > Admin Settings > Line Configuration > Line x > Line Keys > Num Line Keys . | |

Example Configuration

The following illustration shows you how to enable four line keys with the same registered line address. In this example, four line keys are configured with registration address 2062.



The phone will display the registered line address 2062 on four line keys, as shown next.



Enabling 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, you can place one call on hold, switch to another call on the same registered line, and have both calls display. As shown in [Table 6-3: Enabling Multiple Call Appearances](#), you can set the maximum number of concurrent calls per registered line and the default number of calls per line key.

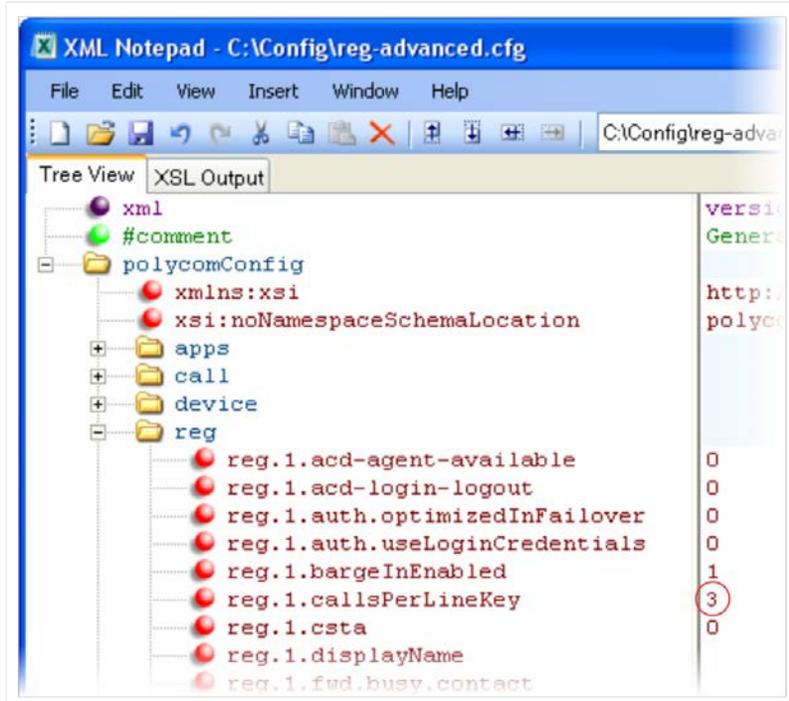
This feature is one of several features associated with *Flexible Call Appearances*. If you want to enable multiple line keys per registration, see [Assigning Multiple Line Keys Per Registration](#). Note that if you assign a registered line to multiple line keys, the default number of concurrent calls will apply to all line keys. If you want use multiple registrations on a phone, see [Enabling Multiple Registrations](#). For definitions of all features associated with Flexible Call Appearances, see [Table 7-4: Flexible Call Appearances](#). Use this table to customize the number of registrations, line keys per registration, and concurrent calls.

Table 7-3: Enabling Multiple Call Appearances

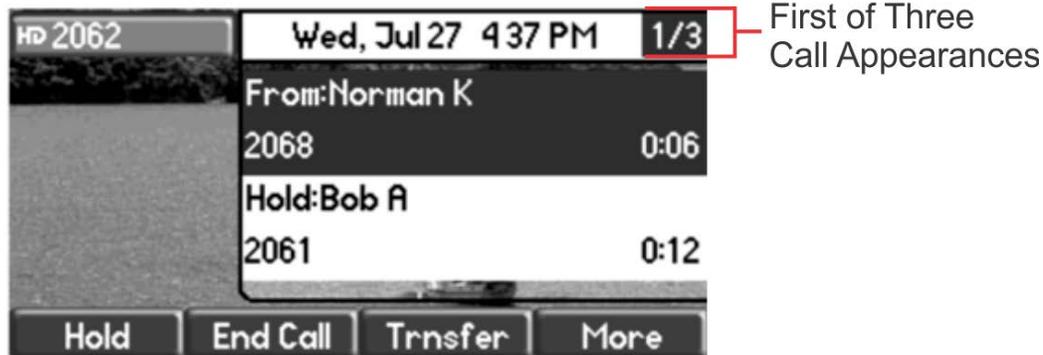
| Central Provisioning Server | template > parameter |
|--|--|
| Set the default number of concurrent calls for all line keys | reg-basic.cfg > call.callsPerLineKey |
| Override the default number of calls per line key for a specific line | reg-advanced.cfg > reg.x.callsPerLineKey |
| Web Configuration Utility | |
| To set the default number of concurrent calls a line key, navigate to Settings > SIP , expand Local Settings , and edit Calls Per Line Key . | |
| To override the number of concurrent calls for a specific line, navigate to Settings > Lines , select the line to modify from the left pane, expand Identification , and edit Calls Per Line . | |
| Local Phone User Interface | |
| Assign the default number of concurrent calls per line by navigating to Menu > Settings > Advanced > Admin Settings > Line Configuration > Calls Per Line Key (navigate to Line Configuration > Line X > Line Keys > Calls Per Line Key to change the calls per line for only line x). | |

Example Multiple Call Appearances Configuration

The following illustration shows that in the **reg-advanced.cfg** template you can enable line 1 on your phone with three call appearances.



Once you have set the `reg.1.callsPerLineKey` parameter to three, you can have three call appearances on line 1. By default, additional incoming calls will be automatically forwarded to your voicemail. If you have more than two call appearances, a call appearance counter will display at the top right corner of your phone's screen as shown next.



A number of features are associated with *Flexible Call Appearances*. Use [Table 7-4: Flexible Call Appearances](#) to understand how you can organize registrations, line keys per registration, and concurrent calls per line key.

In the following table:

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 (includes Conference Legs) The runtime maximum number of concurrent calls. (The number of conference participants minus the moderator.)

The SoundPoint IP 450, 550, 560, and 650 phones support four-way local conferencing.

Table 7-4: Flexible Call Appearances

| Phone Model | Registrations | Line Keys | Line keys Per Reg | Calls Per Line Key | Concurrent Calls* |
|---------------------------------|---------------|-----------|-------------------|--------------------|-------------------|
| SoundPoint IP 321, 331, and 335 | 2 | 2 | 2 | 4 | 8 (2) |
| SoundPoint IP 450 | 3 | 3 | 3 | 8 | 24 (2) |
| SoundPoint IP 550, 560 | 4 | 4 | 4 | 24 | 24 (2) |
| SoundPoint IP 650 | 34 | 48 | 24 | 24 | 24 (3) |
| SoundStation IP 5000 | 1 | na | na | 8 | 8 (2) |
| SoundStation IP 6000 | 1 | na | na | 8 | 8 (2) |
| SoundStation IP Duo | 1 | na | na | 8 | 8 (2) |
| VVX 1500 | 24 | 29 | 24 | 24 | 24 (2) |
| VVX 500 | 12 | 12 | 12 | 24 | 24 (2) |
| SpectraLink 84xx | 6 | 6 | 6 | 24 | 24 (2) |

*Note that each conference leg counts as one call. The total number of concurrent calls in a conference indicated in this table includes all conference participants *minus* the moderator.

Customizing and Downloading Fonts

You can customize the fonts that display on the phone's user interface. Polycom recommends that you use existing fonts embedded in the software as external fonts may not be compatible with the phones. Use the parameters in [Table 7-5: Customizing Fonts](#) to set custom fonts. External fonts must be saved as a Microsoft **.fnt** file format. You can also download fonts external to the existing fonts embedded in the software and load them to the phone. Before you configure or download fonts, familiarize yourself with the guidelines on downloading and loading font files in [](#).



Note: Some Phones Do Not Support Custom Fonts

Custom fonts are not supported on the SoundPoint IP 450, the VVX 500 and 1500, or the SpectraLink handsets.

Table 7-5: Customizing Fonts

| Central Provisioning Server | template > parameter |
|---|--------------------------|
| Specify the name of the font file to load to the phone..... | region.cfg > font.x.name |

Setting the Phone Language

You can select the language that displays on the phone using the parameters in [Table 7-6: Setting the Phone Language](#). Each language is stored as a language file in the **SoundPointIPLocalization** folder. This folder is included with the Polycom UC Software you downloaded to your provisioning server. If you want to edit the language files, you will need to use a Unicode-compatible XML editor such as XML Notepad 2007 and familiarize yourself with the guidelines on basic and extended character support, see [<ml/>](#).

The phones support major western European languages.

All phones except the SoundPoint IP models 321, 331, and 335 support the following the following languages: Simplified Chinese, Traditional Chinese, Danish, Dutch, English, French, German, Italian, Japanese, Korean, Norwegian, Polish, Brazilian Portuguese, Russian, Slovenian, International Spanish, and Swedish.

The SoundPoint 321, 331, and 335 support all languages except Japanese and Korean.



Note: Multilingual Support for the Updater

At this time, the Updater is available in English only.

Table 7-6: Setting the Phone Language

| Central Provisioning Server | template > parameter |
|--|-------------------------------|
| Obtain the parameter value for the language you want to display on the phone | site.cfg > lcl.ml.lang.menu.* |
| Specify the language used on the phone's display screen..... | site.cfg > lcl.ml.lang |

Web Configuration Utility

To change the language of the phone's display screen, navigate to **Simple Setup**, expand **Language**, and change **Phone Language**.

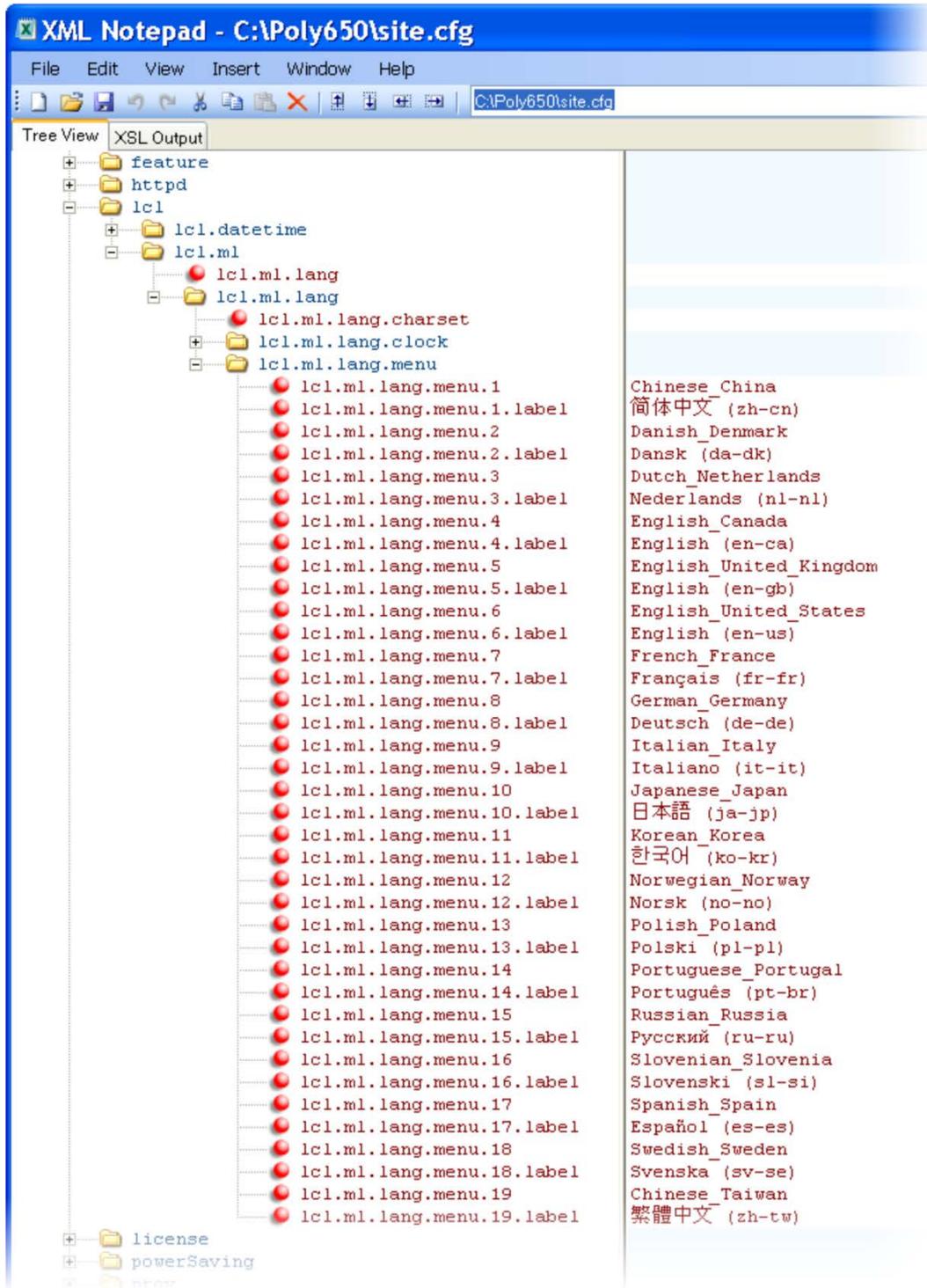
Local Phone User Interface

To change the language of the phone's display screen, navigate to **Menu > Settings > Basic > Preferences > Language**.

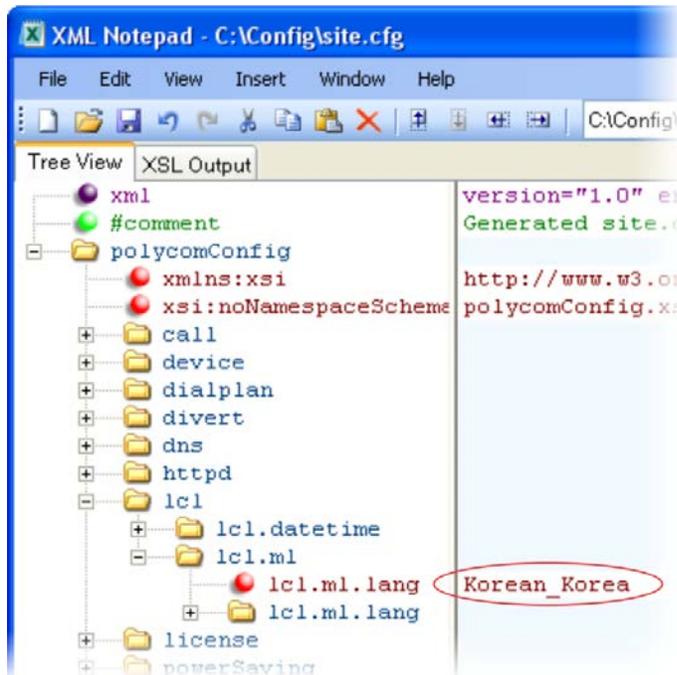
Example Phone Language Configuration

The following illustration shows you how to change the phone language.

Locate the language you want the phone to display in the **site.cfg** template in `lcl.ml.lang.* menu`, as shown next.



From the list, select the language you want to use and enter it in `lcl.ml.lang`. In the following example, the phone is set to use the Korean language.



Once configured, the phone will use Korean characters.



Enabling Instant Messaging

All phones (except the SoundPoint IP 321/331/335) can send and receive instant text messages. You can use the SpectraLink 8400 Series handsets to send and receive instant messages only when integrated with [Setting Up Microsoft Office Communications Server 2007 R2 Integration](#) or [Setting Up Microsoft Lync Server 2010 Integration](#). See [Table 7-7: Enabling Instant Messaging](#) for the parameter you need to set to enable instant messaging. Once the feature is enabled, the phone's message waiting indicator (MWI) LED will alert you to incoming text messages visually; you can also set audio alerts. When you want to send an instant message, you can use the phone's dial pad to type your messages or you can choose a short message from a preset list. You can send instant messages by initiating a new dialogue or by replying to a received message. In addition, you can choose the message destination manually

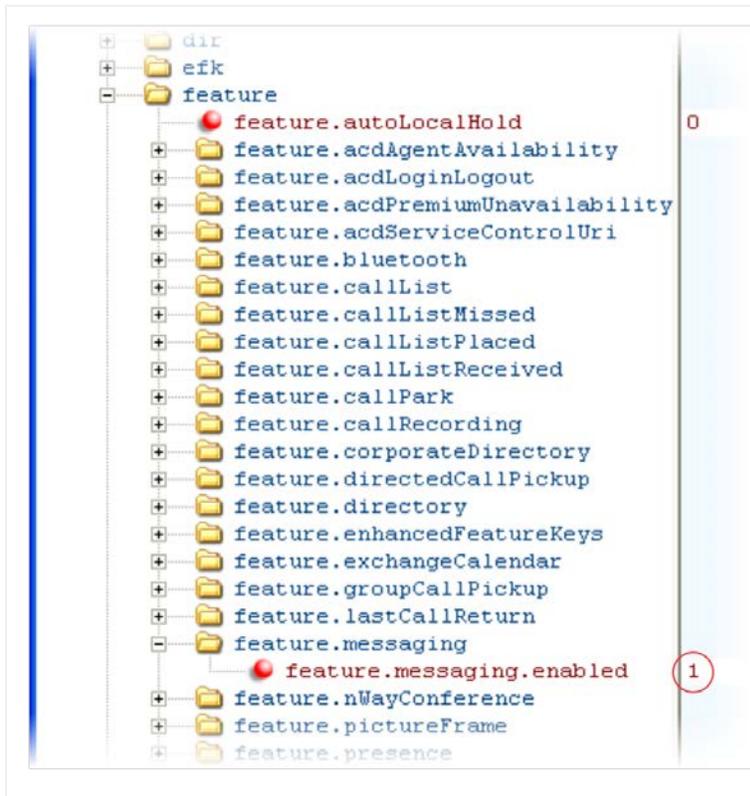
or you can select a contact from your local contact directory (see [Using the Local Contact Directory](#)).

Table 7-7: Enabling Instant Messaging

| | |
|---|--|
| Central Provisioning Server | template > parameter |
| Enable or disable instant messaging | features.cfg > feature.messaging.enabled |

Example Instant Messaging Configuration

The following illustration shows you how to enable instant messaging in the **features.cfg** template.



After setting this parameter, press the **Messages** key on the phone's keypad to display the **Instant Messages** option, as shown next.



Press the **Select** soft key to open the *Instant Messages* menu where you can send and receive instant messages.

Synthesized Call Progress Tones

Polycom phones play call signals and alerts, called call progress tones, such as busy signals, ringback sounds, and call waiting tones. The built-in call progress tones on your phone match standard North American tones. If you would like to customize the phone's call progress tones to match the standard tones in your region, contact Polycom Support.

Using the Microbrowser and Web Browser

The SoundPoint IP 450, 550, 560, and 650 phones, SoundStation IP 5000 and 6000 phones, support an XHTML microbrowser. The VVX phones and SpectraLink handsets support a full Web browser. The microbrowser and browser parameters you can configure are listed in [Table 7-8: Using the Microbrowser](#). Note that the exact functions and performance of the microbrowser and Web browser vary with the model of phone you are using.

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 you can launch in a Web browser by pressing the **Applications** key on the phone or by navigating to **Menu > Applications**. On the SpectraLink handsets, you can launch the Web browser from the Home screen by selecting **Applications**. On the VVX 1500 phone, you can launch the Web browser by pressing the **App** key on the phone or by navigating to **Menu > Applications**. On the VVX 500 phone, go to **Menu > Applications**. On the VVX only, when you tap on a link that displays on the idle browser the phone will launch that link in the Web browser.

Polycom provides a default microbrowser and browser feature for the phone's idle screen. *My Info Portal* is a Polycom-developed application that gives you access to the latest news, sports, weather, stock, and other news. You can sign up for access to [My Info Portal](#) through the Polycom VVX 1500 phone or through a computer. Note that the first time you sign in to *My Info Portal*, you will be asked to accept the Polycom End User Licensing Agreement (EULA).



Note: My Info Portal May Require Browser Setting Changes

To get the *My Info Portal* to appear in the VVX 1500 phone's idle browser, set `mb.idleDisplay.home` to `http://idle.myinfoportal.apps.polycom.com/idle` and `mb.idleDisplay.refresh` to 600.



Note: Web Browser Will Restart

If the browser uses over 30MB of memory and either the amount of free memory on the phone is below 6MB or the real time is between 1am to 5am, the browser will restart. Once the browser has restarted, the last displayed Web page is restored.

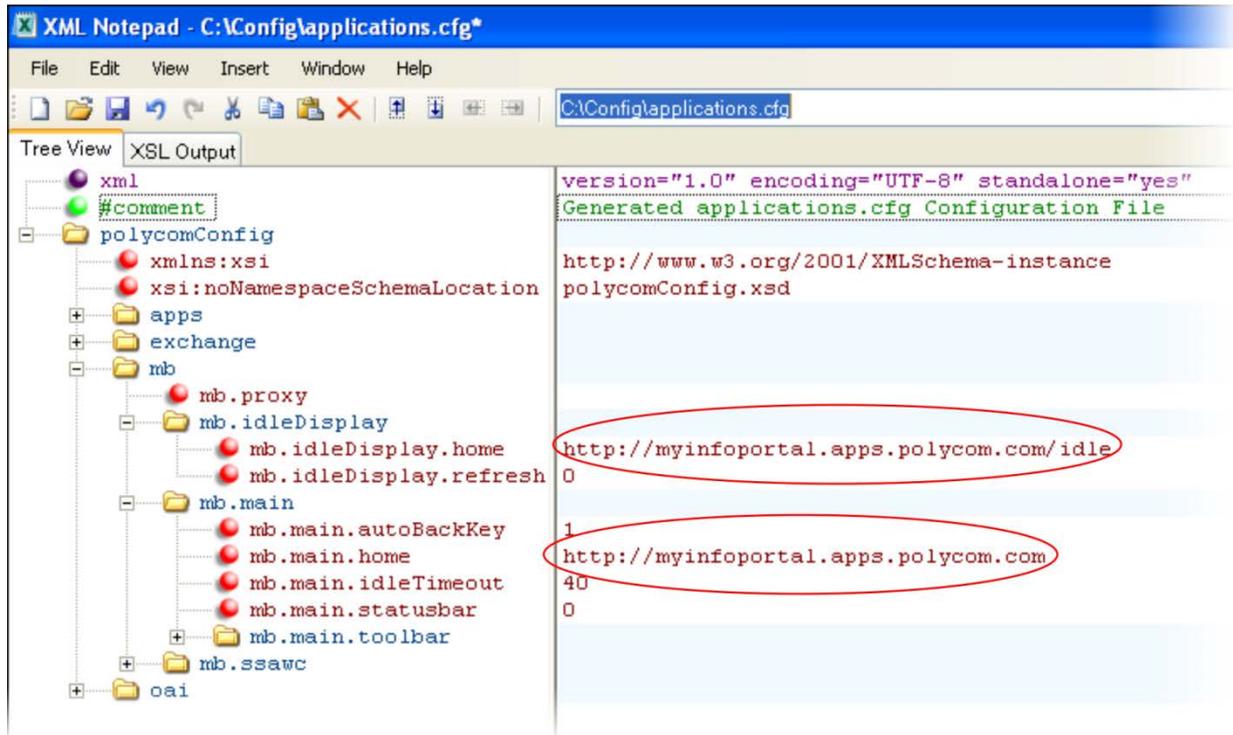
For more information, see the Polycom [Web Application Developer's Guide](#).

Table 7-8: Using the Microbrowser and the Web Browser

| Central Provisioning Server | template > parameter |
|---|---|
| Specify the Application browser home page, a proxy to use, and size limits | applications.cfg > mb.* |
| Specify the Telephony Event Notification events to be recorded and the URL where notifications will be sent | applications.cfg > apps.telNotification.* |
| Specify phone state polling settings, such as response mode, the poll URL, and a user name and password | applications.cfg > apps.statePolling.* |
| Specify the push server settings, including message type, port, tunnel, and a user name and password | applications.cfg > apps.push.* |
| <hr/> | |
| Web Configuration Utility | |
| To specify the Applications browser home page and proxy to use, navigate to Settings > Microbrowser . | |
| To configure telephony event notifications, phone state polling settings, and push settings, navigate to Settings > Applications and see Telephony Event Notification , Phone State Polling , and Push . | |

Example Microbrowser and Web Browser Configuration

The following example shows you how to set a Web page on the idle screen of the VVX phone and how to set the interactive Web browser's home page on the VVX 1500 phone.



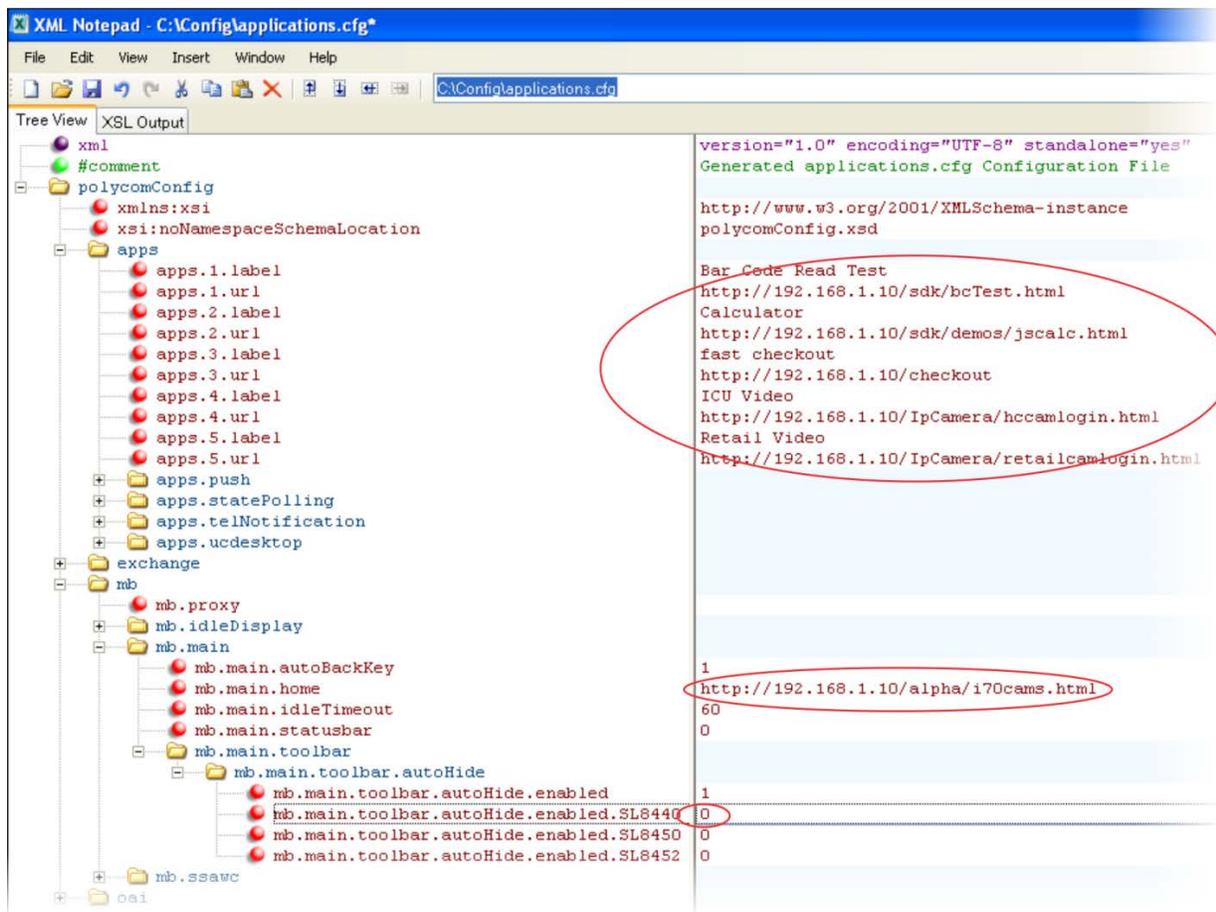
The following illustration shows a non-interactive idle Web browser on the VVX 1500 phone.



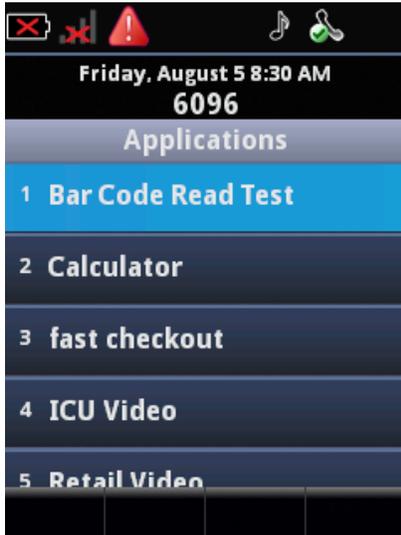
The following illustration shows the Web browser's interactive home page on the VVX 1500 phone.



The following example shows you how to set the interactive Web browser's home page on the SpectraLink handsets.



The following illustration shows the Web browser's interactive home page on the SpectraLink handset.



Configuring Real-Time Transport Protocol Ports

You can configure the phone to filter incoming RTP packets. You can filter the packets by IP address, or by port. For greater security, you can also configure RTP settings to reject packets arriving from a non-negotiated IP address or from an unauthorized source. You can reject packets that the phone receives from a non-negotiated IP address or a non-negotiated port.

You can configure the phone to enforce symmetric port operation for RTP packets. When the source port is not set to the negotiated remote sink port, arriving packets can be rejected.

You can also fix the phone's destination transport port to a specified value regardless of the negotiated port. This can be 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.

You can 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, the next-highest odd-numbered port is used to send and receive RTP. [Table 7-9: Configuring Real-Time Transport Protocol Ports](#) provides a link to the reference section.

The phone is compatible with RFC 1889 - RTP: A Transport Protocol for Real-Time Applications - and the updated RFCs 3550 and 3551. Consistent with RFC 1889, the phone treats all RTP streams as bi-directional from a control perspective and expects that both RTP end points will negotiate the respective destination IP addresses and ports. This allows real-time transport control protocol (RTCP) to operate correctly even with RTP media flowing in only a single direction, or not at all.

Table 7-9: Configuring Real-Time Transport Protocol Ports

| Central Provisioning Server | template > parameter |
|---|--|
| Filter RTP packets by IP address | site.cfg > tcpIpApp.port.rtp.filterByIp |
| Filter RTP packets by port | site.cfg > tcpIpApp.port.rtp.filterByPort |
| Force-send packets on a specified port..... | site.cfg > tcpIpApp.port.rtp.forceSend |
| Set the starting port for RTP packet port range | site.cfg > tcpIpApp.port.rtp.mediaPortRangeStart |

Web Configuration Utility

Filter RTP packets by IP address, by port, force-send packets on a specified port, and set the port range start by navigating to **Settings > Network > RTP**.

Example Real-Time Transport Protocol Configuration

The following illustration shows the default real-time transport protocol settings in the **site.cfg** template file. The parameter `tcpIpApp.port.rtp.filterByIp` is set to 1 so that the phone will reject RTP packets sent from non-negotiated IP addresses. The parameter `tcpIpApp.port.rtp.filterByPort` is set to 0 so that RTP packets sent from non-negotiated ports will not be rejected. Enter a value in the `tcpIpApp.port.rtp.forceSend` parameter to specify the port that all RTP packets will be sent to and received from. The parameter `tcpIpApp.port.rtp.mediaPortrangeStart` shows the default starting port 2222 for RTP packets. The starting port must be entered as an even integer.



Configuring Network Address Translation

The phone can work with certain types of network address translation (NAT). 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 Real-Time Transport Protocol (RTP) traffic use symmetric ports. You can configure the external IP address and ports used by the NAT on the phone's behalf on a per-phone basis. [Table 7-10: Network Access Translation](#) lists each of the parameters you can configure. Note that the source port in transmitted packets is the same as the associated listening port used to receive packets.

Table 7-10: Network Access Translation

| Central Provisioning Server | template > parameter |
|--|--|
| Specify the external NAT IP address | sip-interop.cfg > nat.ip |
| Specify the external NAT keepalive interval | sip-interop.cfg > nat.keepalive.interval |
| Specify the external NAT media port start | sip-interop.cfg > nat.mediaPortStart |
| Specify the external NAT signaling port..... | sip-interop.cfg > nat.signalPort |
| Web Configuration Utility | |
| Specify the external NAT IP address, the signaling port, the media port start, and the keepalive interval by navigating to Settings > Network > NAT . | |

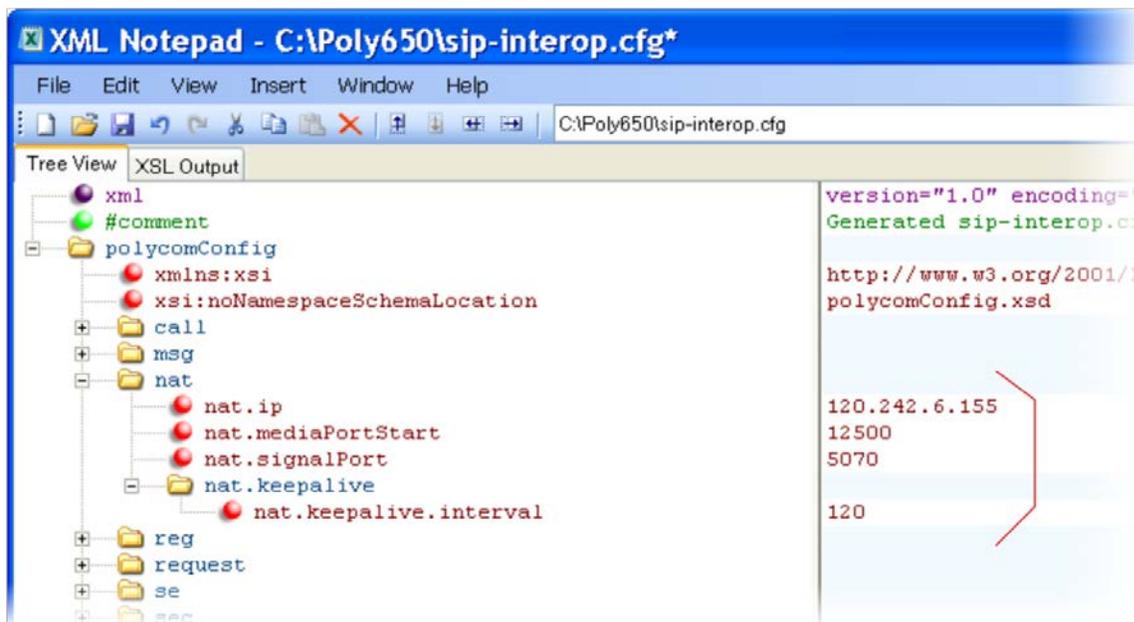
Example Network Address Translation Configuration

The following illustration shows the default NAT parameter settings. The parameter `nat.ip` is the public IP that you want to advertise in SIP signaling. The default IP is 120.242.6.155.

The parameter `nat.mediaPortStart` is the RTP used to send media. If non-Null, this attribute will set the initially allocated RTP port and will override the value set in `tcpIpApp.port.rtp.mediaPortRangeStart`. In the example below, the starting port is 12500 and the phone will cycle through start-port + 47 for phones that support audio only or start-port + 95 for phones that support video.

The parameter `nat.signalPort` specifies the port that the phone will use for SIP signaling. This parameter will override `voIpProt.local.Port`. In the example below, the phone will use port 5070 for SIP traffic.

Use the `nat.keepalive.interval` to specify the keepalive interval in seconds. This parameter sets the interval at which phones will send a keepalive packet to the gateway/NAT device. The keepalive packet keeps the communication port open so that NAT can continue to function as initially set up. In the example below, the phone will send the keepalive every 120 seconds.



Using the Corporate Directory

You can connect your phone to a corporate directory server that supports the Lightweight Directory Access Protocol (LDAP) version 3. The corporate directory is a flexible feature and [Table 7-11: Using the Corporate Directory](#) links you to the parameters you can configure. Once set up on the phones, the corporate directory can be browsed or searched. You can call numbers and save entries you retrieve from the LDAP server to the local contact directory on the phone.

Polycom phones currently support the following LDAP servers:

- Microsoft® Active Directory 2003 SP2
- Sun ONE Directory Server 5.2 p6
- Open LDAP Directory Server 2.4.12
- Microsoft Active Directory Application Mode (ADAM) 1.0 SP1

Polycom phones support corporate directories that support server-side sorting and those that do not. For phones that do not support server-side sorting, sorting is performed on the phone.



Tip: Better Performance With Server-Side Sorting

Polycom recommends using 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](#).



Web Info: Supported LDAP Directories

Configuration of a corporate directory depends on the LDAP server you use. For detailed explanations and examples of all currently supported LDAP directories, see [Technical Bulletin 41137: Best Practices When Using Corporate Directory on Polycom Phones](#).

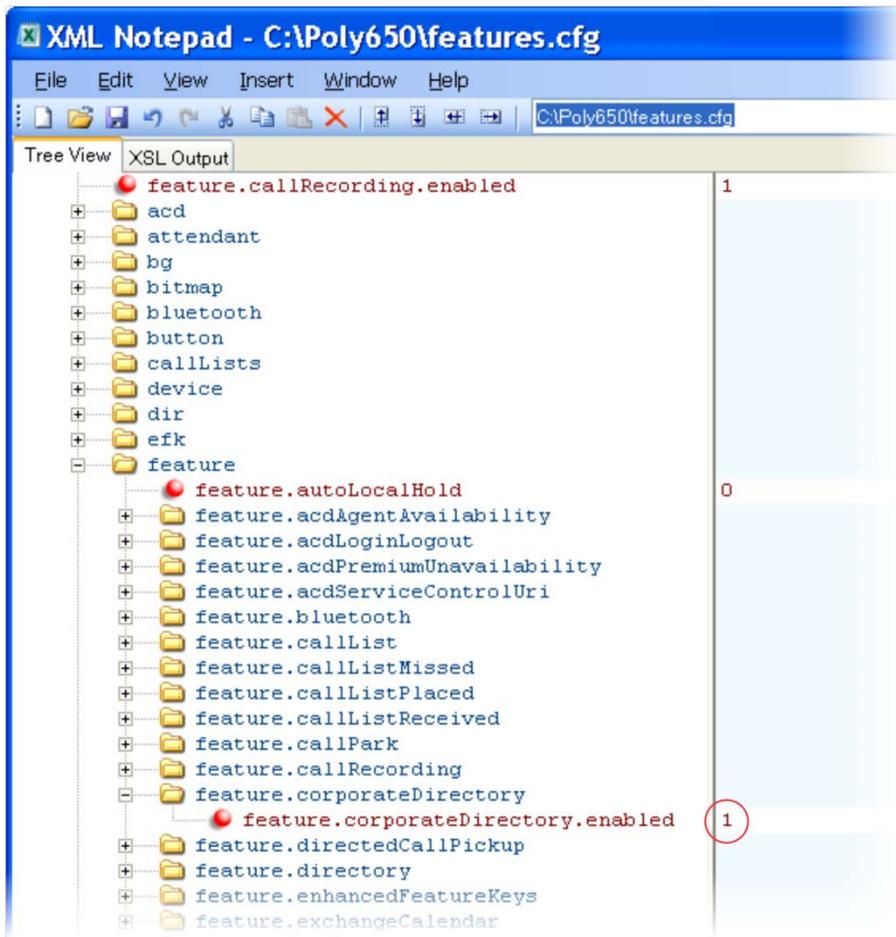
Table 7-11: Using the Corporate Directory

| Central Provisioning Server | template > parameter |
|--|--------------------------------|
| Specify the location of the corporate directory's LDAP server, the LDAP attributes, how often to refresh the local cache from the LDAP server, and other settings..... features.cfg > dir.corp.* | |
| Local Phone User Interface | |
| Specify if the corporate directory should remember the previous search filter by navigating to Settings > Basic > Preferences > Corporate Directory > View Persistency . | |
| Review the corporate directory LDAP server status by navigating to Menu > Status > CD Server Status . | |
| To search your corporate directory, press the Directories key on the phone, and select Corporate Directory . | |

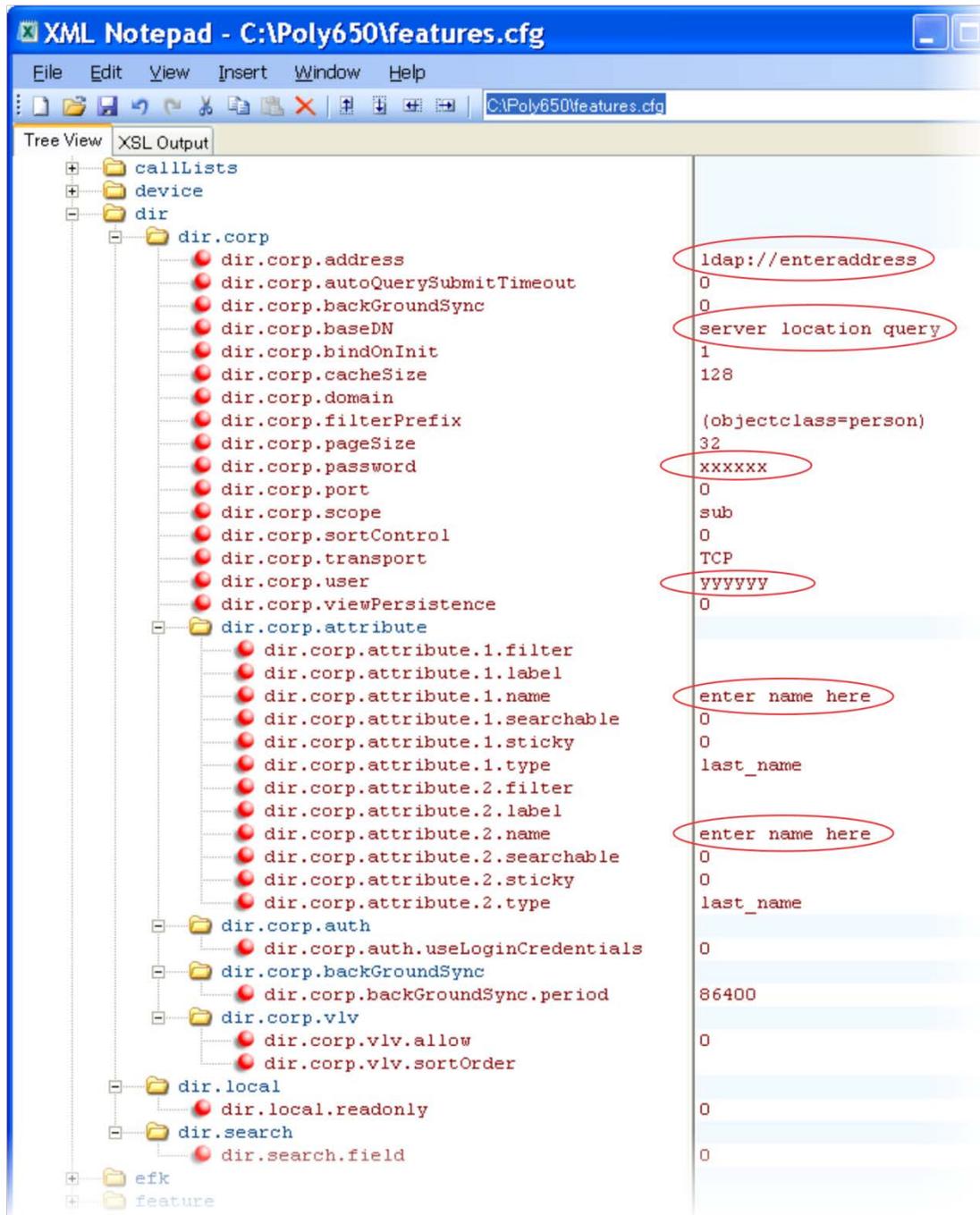
Example Corporate Directory Configuration

The following example is a representation of the minimum parameters you will need to set to begin using the corporate directory. The exact parameters and values you will need to configure vary with the corporate directory you are using.

First, enable the corporate directory feature in the **features.cfg** template, as shown next.



The following illustration points you to the minimum parameters you need to set. You will need to enter a corporate directory address in `dir.corp.address`. You will need to specify where on the corporate directory server you want to make queries in `dir.corp.baseDN`. In addition, you will require a user name and password. The `dir.corp.attribute.x.name` must match the attributes in the server.



To search the corporate directory, press the **Directories** key on the phone and select **Corporate Directory**, as shown next.



CMA Directory

The VVX 1500 Business Media phones have access to the Polycom® Converged Management Application™ (CMA™) system directory, a corporate contact directory stored on the CMA server (also known as the LDAP server). You can search the CMA directory and dial and save entries retrieved from the CMA server to the Buddies list on your phone. You can place phone calls to numbers retrieved from the CMA directory. You can also group CMA Contacts on the CMA Server.

To access the CMA directory, the VVX phone must be provisioned using the Polycom CMA system. For information on provisioning VVX phones using the Polycom CMA system, see [Provisioning VVX Phones Using a Polycom CMA System](#) in Chapter 4 of this guide.

The CMA Directory interface is read only and you cannot add, edit, or delete directory entries. Note that the Polycom CMA system looks up and displays the name on incoming calls only for lines registered to H.323.



Web Info: Using the CMA System

For details on how to use the CMA system with the VVX 1500 phones, refer to the section *Working with a Polycom CMA System* in the [User Guide for the Polycom VVX 1500 Business Media Phone](#).



Tip: Using the CMA Directory on the VVX 1500

The CMA directory is available only on the VVX 1500 phone. In order to use the CMA directory, you will need to provision the phone using the Polycom CMA system.

Recording and Playing Audio Calls

You can configure the SoundPoint IP 650 and the VVX phones to record audio calls to a USB device that you plug into the phone. You can play back recorded audio on the phone as well as on other devices that run applications like Windows Media Player® or iTunes® on a Windows®- or Apple®-based computer.

To enable this feature, the USB device must be compatible with Polycom phones.



Web Info: Supported USB Devices

For a list of supported USB devices, see [Technical Bulletin 38084: Supported USB Devices for Polycom SoundPoint IP 650 and VVX Phones](#).

You can enable call recording with the parameter shown in [Table 7-12: Recording and Playing Audio Calls](#). Audio calls are recorded in **.wav** format and include a date/time stamp, for example, **20Apr2007_190012.wav** was created on April 20, 2007 at 19:00:12. The phone will display the recording time remaining on the attached USB device and you can browse all recorded files using the phone’s menu.



Settings: Using Call Recording with the Polycom IP 650 Phone

By default, the call recording feature is disabled. When running Polycom UC Software 4.1.x and you enable call recording on the Polycom IP 650 phone, you are advised to try to limit the maximum number of concurrent calls to about 12 due to memory consumption of the call recording feature. If call recording is disabled, you can use the maximum number of 24 concurrent calls, which is the default limit.



Note: Informing Parties When You Are Recording calls

Federal, state, and/or local laws may legally require that you to notify some or all of the call parties that you are recording.

Table 7-12: Recording and Playing Audio Calls

| | |
|---|--|
| Central Provisioning Server | template > parameter |
| To enable or disable call recording | features.cfg > feature.callRecording.enabled |

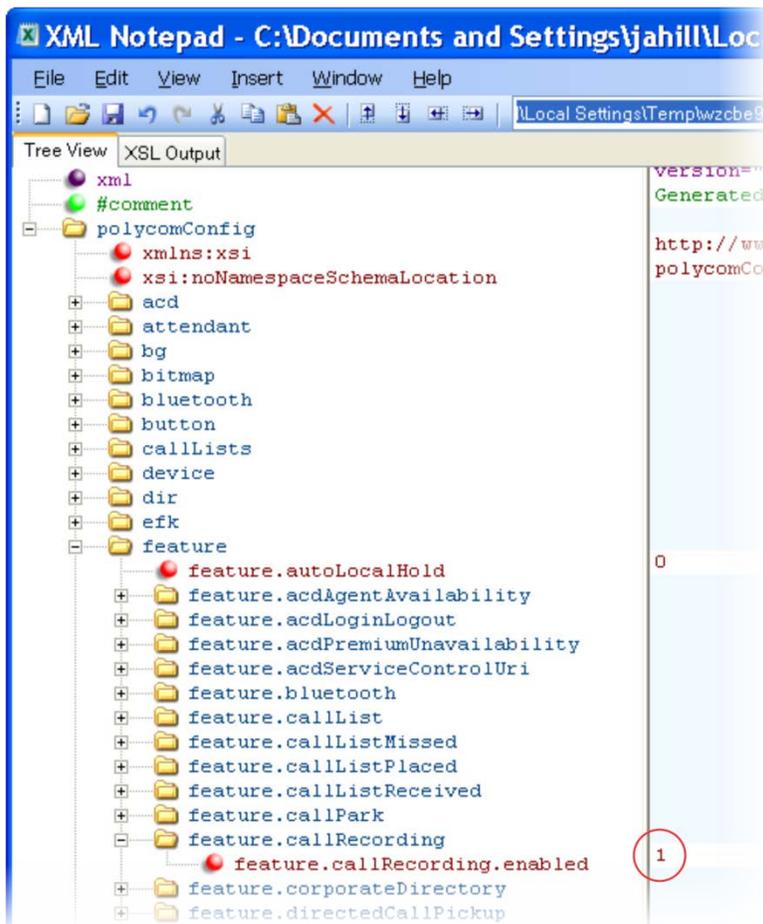
Local Phone User Interface

Browse your audio files by navigating on the phone to **Menu > Features > Removable Storage Media > Browse Recordings**.

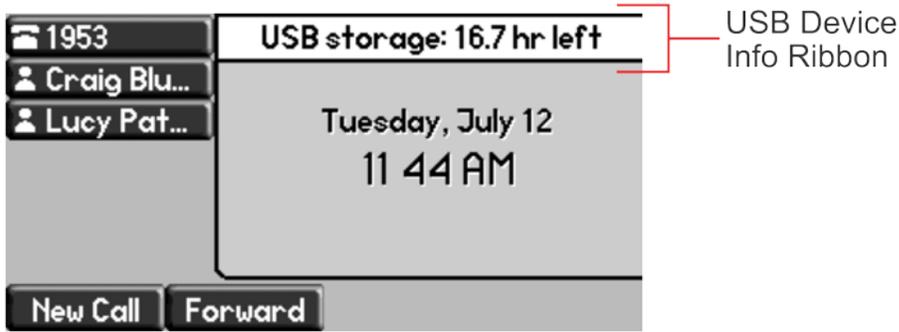
View the properties of your USB device including size, available space, and remaining recording time by navigating on your phone to **Menu > Features > Removable Storage Media > Storage Media Properties**.

Example Call Recording Configuration

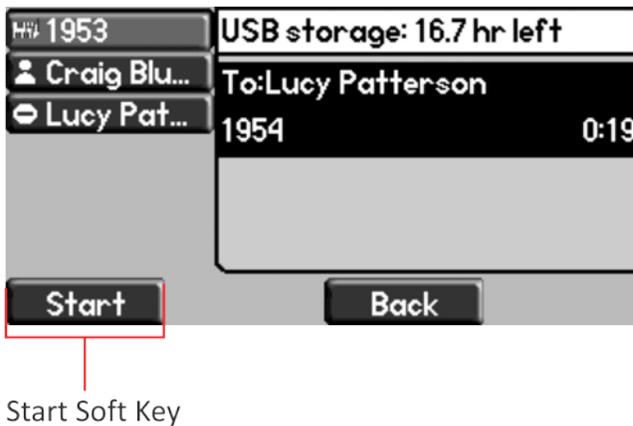
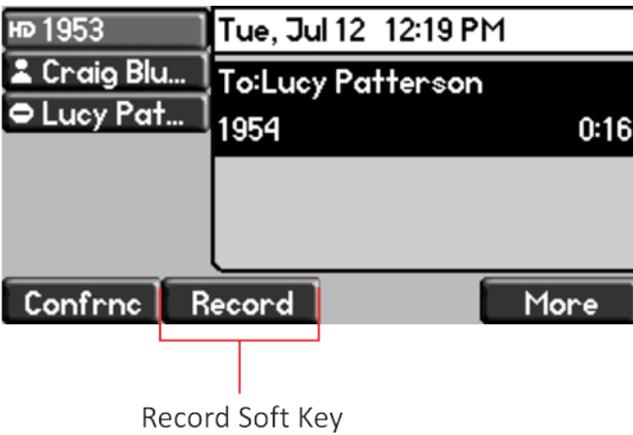
To record audio from the phone, you will need a USB device plugged into the phone, and you will need to enable the call recording feature in the **features.cfg** template file. In **features.cfg**, you will need to locate `feature.callRecording.enabled` and enter '1', as shown next.



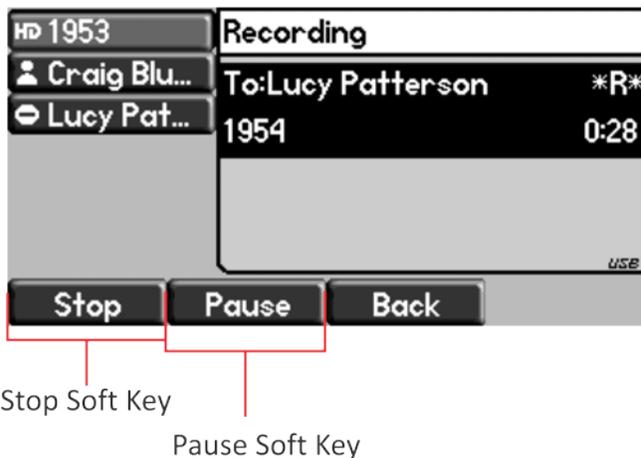
Plug the USB device into the phone. When a compatible USB device is plugged into the phone, a USB info ribbon displays on the phone's screen when the phone is in the idle state, shown next:



When you begin an active call, a **Record** soft key displays on the phone screen. If you want to record an audio call, press the **Record** soft key to display a **Start** soft key, shown next.



Pressing the **Start** soft key starts recording audio. A **Pause** and a **Stop** soft key will display, as shown next.



Press the **Pause** soft key to pause recording and press the **Stop** soft key to stop recording.

You can browse recorded audio files by navigating on the phone to **Menu > Removable Storage Media > Browse Recordings**.

Configuring the Digital Picture Frame

On the VVX phones you can display a slide show of images on the phone's idle screen. Images must be saved in JPEG, BMP, or PNG format on a directory on a USB device that is attached to the phone. The parameters you can configure are listed in [Table 7-13: Configuring the Picture Frame](#). The phone can display a maximum image size of 9999x9999 pixels and a maximum of 1000 images.



Note: Maximum Image Size

Although 9999x9999 images and progressive/multiscan JPEG images are supported, the maximum image size that can be downloaded is restricted by the available memory in the phone.

Table 7-13: Configuring the Picture Frame

| Central Provisioning Server | template > parameter |
|---|---|
| To enable or disable the digital picture frame..... | features.cfg > feature.pictureFrame.enabled |
| Specify the name of the folder on the USB device containing the images
..... | reg-advanced.cfg > up.pictureFrame.folder |
| Set how long each picture will display | reg-advanced.cfg > up.pictureFrame.timePerImage |

Web Configuration Utility

To specify the name of the folder containing the images and the time for each image to display, navigate to **Preferences > Additional Preferences** and expand **Picture Frame Settings**.

Local Phone User Interface

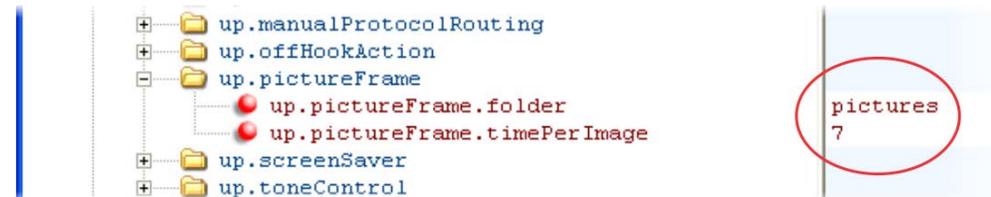
To specify the name of the folder containing the images and the time for each image to display, navigate to **Menu > Settings > Basic > Preferences > Picture Frame**.

Example Digital Picture Frame Configuration

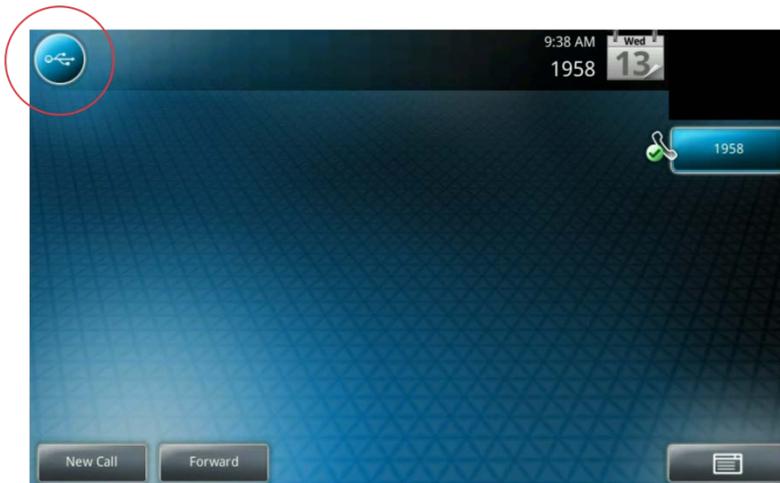
In the following illustration, the digital picture frame feature is enabled in the **features.cfg** template file.



In the **reg-advanced.cfg** template file, the phone will look on the USB device for images in the folder named *pictures* and each picture will display for 7 seconds.



Once the configuration is complete, restart the phone, insert the USB device to the phone. A Removable Storage Media icon displays on the phone's screen, shown next on the VVX 1500.



To show your pictures, press the icon and press **Picture Frame**.



Note: Accessing the Digital Picture Frame

The digital picture frame can be accessed through the *PicFrame:// URL*.

Configuring Enhanced Feature Keys

Enhanced Feature Keys (EFK) enables you to customize the functions of a phone's line and soft keys and, as of UC Software 4.0.1, hard keys. You can use EFK to assign frequently used functions to line keys, soft keys, and hard keys or to create menu shortcuts to frequently used phone settings.

See [Table 7-14: Configuring Enhanced Feature Keys](#) for the parameters you can configure and a brief explanation of how to use the contact directory to configure line keys. Enhanced feature key functionality is implemented using star code sequences (like *69) 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. The macros are case sensitive.

The rules for configuring EFK for line keys, soft keys, and hard keys are different. Before using EFK, you are advised to become familiar with the macro language shown in this section and in the reference section at [<efk/>](#).



Web Info: Using Enhanced Feature Keys

For instructions and details on how to use Enhanced Feature Keys, refer to [Technical Bulletin 42250: Using Enhanced Feature Keys and Configurable Soft Keys on SoundPoint IP, SoundStation IP, and VVX 1500 Phones](#).

Note that the configuration file changes and the enhanced feature key definitions can be included together in one configuration file. Polycom recommends creating a new configuration file in order to make configuration changes.



Tip: EFK Compatibility

The Enhanced Feature Key (EFK) feature from SIP 3.0 is compatible with Enhanced Feature Key feature from SIP 3.1. However, improvements have been made and Polycom recommends that existing configuration files be reviewed and updated.

Table 7-14: Configuring Enhanced Feature Keys

| Central Provisioning Server | template > parameter |
|---|--|
| Specify at least two calls per line key | reg-basic.cfg > reg.x.callsPerLineKey |
| Enable or disable Enhanced Feature Keys | features.cfg > feature.enhancedFeatureKeys.enabled |
| Specify the EFK List parameters | features.cfg > efk.efklist.x.* |
| Specify the EFK Prompts | features.cfg > efk.efkprompt.x.* |

Because line keys and their functions are linked to fields in the contact directory file - **000000000000-directory.xml** (global) or **<MACaddress>-directory.xml** (per phone) - you will 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. For more detailed information on using the contact directory, see [Using the Local Contact Directory](#)..... **000000000000-directory~.xml**

Some Guidelines for Configuring Enhanced Feature Keys

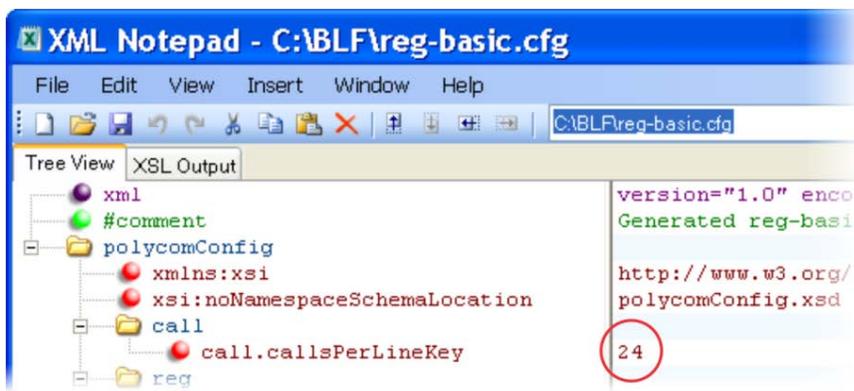
The following guidelines will help you to configure EFK efficiently:

- Activation of EFK functions requires valid macro construction.
- All failures are logged at level 4 (minor).
- If two macros have the same name, the first one will be used and the subsequent ones will be 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 “^” so it will 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, “!1!2”).
- A sequence of macro names and speed dial references is supported (for example, “!m1!2!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 once you reach 50 macro substitutions).

Enhanced Feature Key Examples

The following illustration shows the default value 24 calls per line key. Ensure that you specify at least two calls per line key.

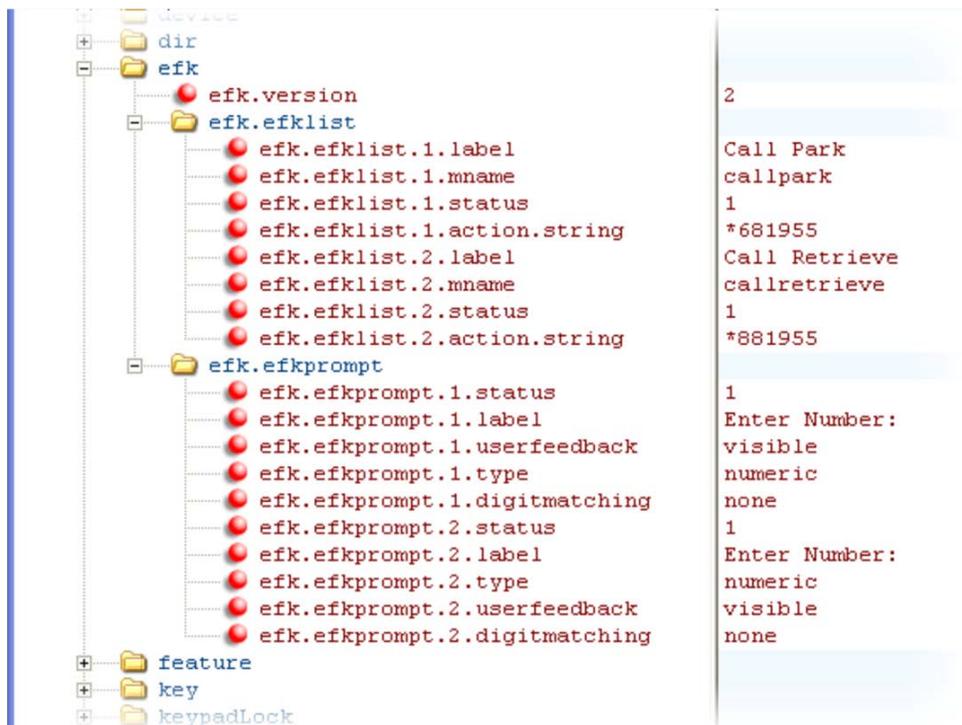


Enable the enhanced feature keys feature in the **features.cfg** template file, as shown next.



In the following illustration, the EFK parameters are located in the **features.cfg** template file. In the `efk.efklist.x.*` parameters, line key '1' has been assigned a Call Park address (1955) and line key '2' a Call Retrieve function. The parameter `acton.string` shows you the macro definition for these two functions. In addition, `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.

In the `efk.prompt.*` parameters, `status` has been enabled. The label on the user prompt has been defined as *Enter Number:* and this prompt will display on the phone screen. The `type` parameter has been set to `numeric` to allow only numbers and because `userfeedback` has been specified as `visible`, you will be able to see the numbers you enter into the prompt.



Understanding Macro Definitions

The `efk.efklist.x.action.string` can be defined by one of the following:

- [Macro Action](#)
- [Prompt Macro Substitution](#)
- [Expanded Macros](#)

Macro Action

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 following fields.

Table 7-15: Macro Actions and Descriptions

\$L<label>\$

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 will be 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 this label is used; otherwise another label may be used and this one ignored.

digits

The digits to be sent. The appearance of this parameter depends on the action string.

\$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

\$T<type>\$

The embedded action type. Multiple actions can be defined. Supported action types include:

- invite
- dtmf
- refer

Note: Polycom recommends that you always define this field. If it is not defined, the supplied digits will be 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 is not defined, the execution of the action string fails.

\$P<prompt num>N<num digits>\$

The user input prompt string. See [Prompt Macro Substitution](#).

\$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 function. For more information, see [Internal Key Functions](#).

URL

A URL. Only one per action string is supported.

Prompt Macro Substitution

The `efk.efklist.x.action.string` can be defined by a macro substitution string, **PnNn** where:

- 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 will fail. The user needs to press the **Enter** soft key to complete data entry.

The macros provide a generic and easy to manage way to 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 once all user input has been collected. The 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.

Expanded Macros

Expanded macros are prefixed with the ^ character and are inserted directly into the local directory `contact` field. For more information, see [Using the Local Contact Directory](#).

Special Characters

The following special characters are used to implement the enhanced feature key functionality. Macro names and macro labels cannot contain these characters. If they do, you may experience unpredictable behavior.

- ! 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).

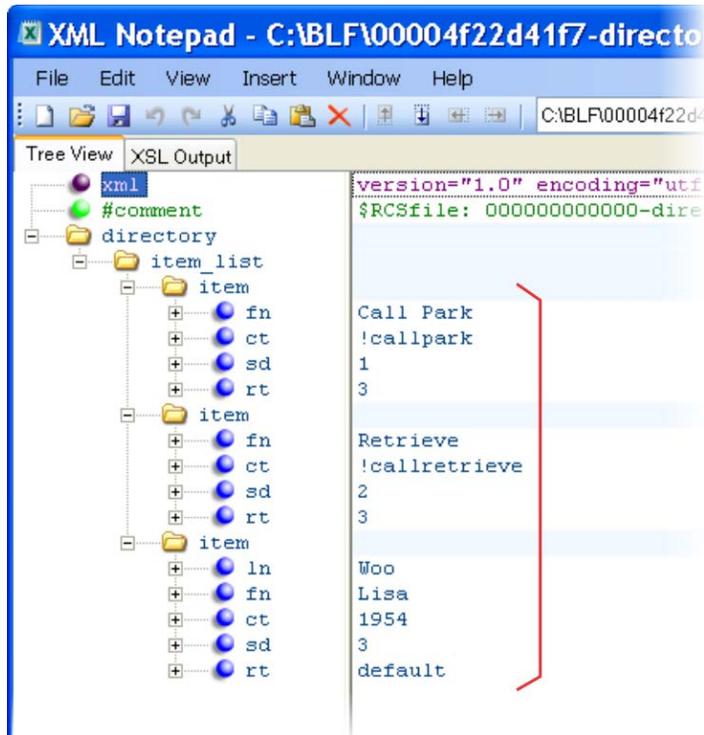
Example Macro

The action string:

`$Changup$*444*$P1N4$$Tinvite$$Cwaitconnect$$P2N3$$Cpause2$$Tdtmf$$Changup$` is executed in order as follows:

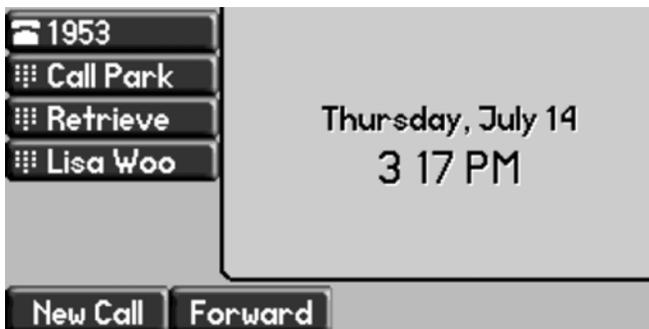
- 5 The user is prompted for 4 digits. For example, *1234*.
- 6 The user is prompted for 3 digits. For example, *567*.
- 7 The user's active call is disconnected.
- 8 The string **444*1234* is sent using the INVITE method.
- 9 Once connected, there is a 2 second pause, and then the string *567* is sent using DTMF dialing on the active call.
- 10 The active call is disconnected.

Because line keys and their functions are linked to fields in the directory file, a macro name you enter in `efk.list.x.mname` must match the name you enter to the `contact (cn)` field in the directory file. The macro name you enter in the (ct) field of the directory file must begin with the '!' prefix. The following example directory file shows a line key configured with Call Park, Call Retrieve, and a speed dial contact Lisa Woo.



For an explanation of all fields in the directory file, see [Table 6-12: Understanding the Local Contact Directory](#).

The following illustrates the Call Park and Call Retrieve line keys and a speed dial contact Lisa Woo.



Speed Dial Example

If your organization's voicemail system is accessible through 7700 and your voicemail password is 2154, you can use a speed dial key to access your voicemail by entering `7700$Cpause3$2154` as the contact number in the `contact (ct)` element.

**Tip: Ensuring Users Do Not Delete Definitions in the Contact Directory**

To avoid users accidentally deleting the definitions in the contact directory, make the contact directory read only.

Configuring Soft Keys

You can customize the functions of the phone's soft keys. This feature is typically used to access frequently used functions or to create menu shortcuts to frequently used phone settings. The parameters that configure soft keys are shown in [Table 7-16: Configuring Soft Keys](#). As with EFK line keys, you assign functions to soft keys using macros. For a list of the available macros, see [Understanding Macro Definitions](#) in

[Configuring Enhanced Feature Keys](#). You can configure soft keys on the SoundPoint IP 321/331/335, 450, 550, 560, and 650 phones, the SoundStation IP 5000 and 6000 phones, and VVX phones, and SpectraLink handsets.

You can configure the soft keys to display 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.

Custom soft keys can be added in 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.

On the SpectraLink handsets, you can customize the flyout menu of the Features soft key. On SoundStation IP, SoundStation IP, VVX phones, you can disable the display of any default soft key to make room for custom soft keys. Or, if your phone does not have a particular hard key, you may want to create a soft key. For example, if the phone does not have a **Do Not Disturb** hard key, you can create a **Do Not Disturb** soft key.

New soft keys can be created as:

- An Enhanced Feature Key sequence
- A speed dial contact directory entry

- An Enhanced Feature Key macro
- A URL
- A chained list of actions

The default soft keys that can be disabled include:

- **New Call**
- **End Call**
- **Split**
- **Join**
- **Forward**
- **Directories** (or **Dir** as it is called on the SoundPoint IP 321/331/335)
- **Callers** (displays on the SoundPoint IP 321/331/335)
- **MyStatus** and **Buddies**
- **Hold, Transfer, and Conference**



Note: Inserting Soft Keys Between the **Hold, Transfer, and Conference Soft Keys**

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.

If you want your 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.

Up to 10 custom soft keys can be configured. If more soft keys are configured than fit on the phone's screen, a **More** soft key displays. Press the **More** soft key to view the remaining soft keys.

[Table 7-16: Configuring Soft Keys](#) shows you the parameters for configuring soft keys. However, this feature is part of Enhanced Feature Keys (EFK) and you must enable the enhanced feature keys parameter to configure soft keys. See

[Configuring Enhanced Feature Keys](#) for details about configuring soft keys and line keys on the phone.

Table 7-16: Configuring Soft Keys

| Central Provisioning Server | template > parameter |
|--|---|
| To turn Enhanced Feature Keys on (required) ... | features.cfg > feature.enhancedFeatureKeys.enabled |
| Specify the macro for a line key or soft key function | features.cfg > softkey.x.action |
| To enable a custom soft key | features.cfg > softkey.x.enable |
| Specify the position of the soft key on the phone screen | features.cfg > softkey.x.insert |
| Specify the text to display on the soft key label | features.cfg > softkey.x.label |
| To position the custom soft key before the default soft keys | features.cfg > softkey.x.precede |
| Specify which call states the soft key will display in | features.cfg > softkey.x.use.* |
| To display soft keys for various phone features, including default soft keys | features.cfg
> softkey.feature.* |

Example Soft Key Configurations

This section provides a few examples of available soft key configurations.



Web Info: Using Configurable Soft Keys

For more examples, see [Technical Bulletin 42250: Using Enhanced Feature Keys and Configurable Soft Keys on Polycom Phones](#).

To disable the New Call soft key:

- 1 In the **features.cfg** template file, set `softkey.feature.newcall` to '0'.
- 2 Reboot the phone.

The **New Call** soft key is not displayed and the soft key space it occupied is empty.

To map a chained list of actions to a soft key:

- 1 Configure speed dial index 2 in the contact directory file with a phone address. For example, enter '2900' in the contact (ct) field.
- 2 In the contact directory, enter '!2' in the contact (ct) field of speed dial index 1.
- 3 Update the configuration file as follows:


```
softkey.1.label = ChainAct
softkey.1.action = $S1$Tinvite$
softkey.1.use.idle = 1
```
- 4 Reboot the phone.

A soft key **ChainAct** displays. Press **ChainAct** to dial the phone number 2900.

To map the Do Not Disturb Enhanced Feature Key sequence to a soft key:

- 1 Update the configuration file as follows:

```
softkey.1.label = DND
softkey.1.action = $FDoNotDisturb$
softkey.1.use.idle = 1
```

- 2 Reboot the phone.

A **DND** soft key is displayed on the phone when it is in the idle state. When the **DND** soft key is pressed, the Do Not Disturb icon is displayed.

To map a Send-to-Voicemail Enhanced Feature Key sequence to a soft key:

- 1 Update the configuration file as follows:

```
softkey.2.label = ToVMail
softkey.2.action = ^*55$P1N10$$Tinvite$
softkey.2.use.alerting = 1
```

- 2 Reboot the phone.

When another party calls, the **ToVMail** soft key is displayed. When the user presses the **ToVMail** soft key, the other party is transferred to voicemail.



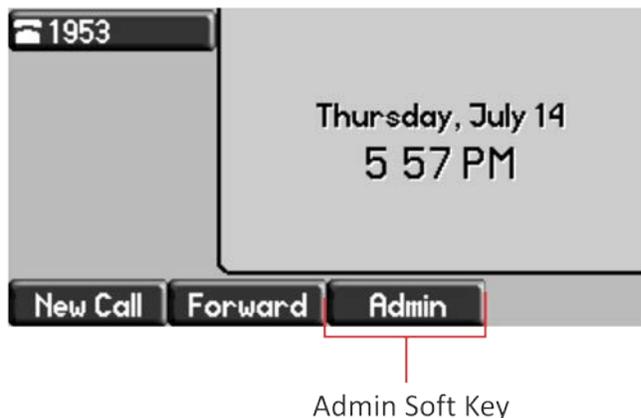
Tip: Active Call Transfer Star Codes Depend On Your Call Server

The exact star code to transfer the active call to Voicemail depends on your call server.

The following example enables a soft key in the phone's idle state that navigates to a phone's administrator settings. The soft is inserted in soft key position 3, after the default soft keys. Note the macro action string:

```
$FMenu$$FDialpad3$$FDialpad2$$FDialpad4$$FDialpad5$$FDialpad6$$FSoftKey1$
```





Enabling the Power Saving Feature

The VVX 500 and 1500 phones support a power-saving feature. This feature has a number of options you can configure, as listed in [Table 7-16: Configuring Soft Keys](#). You can 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. Or, if you want to turn on the power-saving feature while at work, you can configure the sensitivity of the phone's motion detection system and an idle time after which the phone enters the power-saving mode.

Table 7-17: Power Saving

| Central Provisioning Server | template > parameter |
|--|---|
| Turn the power-saving feature on or off..... | site.cfg > powerSaving.enable |
| Specify the amount of time before the phone screen goes idle..... | site.cfg > powerSaving.idleTimeout.* |
| Set the office hour start time and duration for each day of the week | site.cfg > powerSaving.officeHours.* |
| Set the phone's motion detection sensitivity..... | site.cfg > powerSaving.userDetectionSensitivity.* |

Web Configuration Utility

To turn this feature on or off and configure how it works, navigate to **Settings > Power Saving** and expand the panels to set the general, office hour, idle timeout, and user detection sensitivity settings.

Local Phone User Interface

To configure the Power Saving **Office Hours**, **Timeouts**, and **User Detection**, navigate to **Menu > Settings > Basic > Power Saving**.

Example Power-Saving Configuration

The power-saving feature is enabled by default on the VVX 1500. The following illustration shows the power-saving default settings, which reflect the hours of a typical work week.

The screenshot shows the XML Notepad interface with the following configuration for the `powerSaving` element:

```

<powerSaving>
  <powerSaving.enable>1</powerSaving.enable>
  <powerSaving.idleTimeout>
    <powerSaving.idleTimeout.offHours>1</powerSaving.idleTimeout.offHours>
    <powerSaving.idleTimeout.officeHours>10</powerSaving.idleTimeout.officeHours>
    <powerSaving.idleTimeout.userInputExtension>10</powerSaving.idleTimeout.userInputExtension>
  </powerSaving.idleTimeout>
  <powerSaving.motionDetection>
    <powerSaving.motionDetection.frameRate>10</powerSaving.motionDetection.frameRate>
  </powerSaving.motionDetection>
  <powerSaving.officeHours>
    <powerSaving.officeHours.duration>
      <powerSaving.officeHours.duration.friday>10</powerSaving.officeHours.duration.friday>
      <powerSaving.officeHours.duration.monday>10</powerSaving.officeHours.duration.monday>
      <powerSaving.officeHours.duration.saturday>0</powerSaving.officeHours.duration.saturday>
      <powerSaving.officeHours.duration.sunday>0</powerSaving.officeHours.duration.sunday>
      <powerSaving.officeHours.duration.thursday>10</powerSaving.officeHours.duration.thursday>
      <powerSaving.officeHours.duration.tuesday>10</powerSaving.officeHours.duration.tuesday>
      <powerSaving.officeHours.duration.wednesday>10</powerSaving.officeHours.duration.wednesday>
    </powerSaving.officeHours.duration>
    <powerSaving.officeHours.startHour>
      <powerSaving.officeHours.startHour.friday>8</powerSaving.officeHours.startHour.friday>
      <powerSaving.officeHours.startHour.monday>8</powerSaving.officeHours.startHour.monday>
      <powerSaving.officeHours.startHour.saturday>8</powerSaving.officeHours.startHour.saturday>
      <powerSaving.officeHours.startHour.sunday>8</powerSaving.officeHours.startHour.sunday>
      <powerSaving.officeHours.startHour.thursday>8</powerSaving.officeHours.startHour.thursday>
      <powerSaving.officeHours.startHour.tuesday>8</powerSaving.officeHours.startHour.tuesday>
      <powerSaving.officeHours.startHour.wednesday>8</powerSaving.officeHours.startHour.wednesday>
    </powerSaving.officeHours.startHour>
  </powerSaving.officeHours>
  <powerSaving.userDetectionSensitivity>
    <powerSaving.userDetectionSensitivity.offHours>2</powerSaving.userDetectionSensitivity.offHours>
    <powerSaving.userDetectionSensitivity.officeHours>7</powerSaving.userDetectionSensitivity.officeHours>
  </powerSaving.userDetectionSensitivity>
</powerSaving>

```

Configuring Push-to-Talk and Group Paging

The Push-to-Talk (PTT) and Group Paging features are supported on all Polycom phone models installed with UC software 4.0.0 or later.

The Group Paging feature enables you to make pages —one-way audio announcements — to users subscribed to a page `group`. The Push-to-Talk (PTT) feature is a collaborative tool that enables you to exchange broadcasts to users subscribed to a PTT `channel`, much like a walkie-talkie. You can transmit pages and PTT broadcasts using your handset, headset, or speakerphone and you can reject them, place them on hold, and end them at any time. PTT broadcasts can be received on the speakerphone, handset, and headset, and pages can be received only through the speakerphone. Both features are available on all phones that use UC Software 4.0.0 or later.

You can enable one of these features or you can operate both simultaneously. Paging and PTT each have 25 groups/channels you can subscribe to.

- **PTT Mode** PTT mode is intended primarily for Wi-Fi phones such as the SpectraLink handsets. In PTT mode, the phone behaves like a walkie-talkie; you can broadcast audio to a PTT channel and recipients subscribed to that channel can respond to your message. To configure PTT, see
- [Table 7-18: Configuring Push-to-Talk](#) for the parameters.
- **Paging Mode** Paging mode is intended primarily for desktop phones. In Paging mode, you can send announcements to recipients subscribed to a page group. In Page mode, announcements play only through the phone's speakerphone. To configure Paging, see [Table 7-19: Configuring Group Paging](#) for the parameters.

Administrators must enable Paging and PTT before users can subscribe to a page group or PTT channel.



Web Info: Using a Different IP multicast address

The Push-to-Talk and Group Paging features use a 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](#).

Push-to-Talk

You specify the same IP multicast address in the parameter `ptt.address` for both PTT and Paging mode. PTT administrator settings are located in the `site.cfg` template file. PTT channels settings are located in the `features.cfg` template file.



Tip: Compatibility With Earlier SpectraLink Handsets

You can configure the PTT feature to be compatible with the earlier SpectraLink 8020 and 8030 Series Wireless Handsets by setting the `ptt.compatibilityMode` parameter to '1'.

Table 7-18: Configuring Push-to-Talk

| Central Provisioning Server | template > parameter |
|---|---|
| Specify the IP multicast address used for the PTT and paging features | site.cfg > ptt.address |
| Enable PTT mode | site.cfg > ptt.pttMode.enable |
| Specify the name to display (per phone) | site.cfg > ptt.displayName |
| Change default settings for PTT mode | site.cfg > ptt.* |
| Specify settings for all PTT channels | features.cfg > ptt.channel.* |
| <hr/> | |
| Web Configuration Utility | |
| To specify the IP multicast address and port, and available channels for PTT paging, navigate to Settings > Paging/PTT Configuration and expand Settings and PTT Mode Configuration . | |
| <hr/> | |
| Local Phone User Interface | |
| Specify the IP multicast address and port, and available channels for PTT from the Paging/PTT Configuration menu, accessible from Menu > Settings > Advanced > Admin Settings . | |
| Users can access basic PTT settings from Menu > Settings > Basic > Preferences > Paging/PTT Configuration . | |

Group Paging

You specify the same IP multicast address in the parameter `ptt.address` for both PTT and Paging mode. Paging administrator settings are located in the `site.cfg` template file. Page group settings are located in the `features.cfg` template file.

Table 7-19: Configuring Group Paging

| Central Provisioning Server | template > parameter |
|---|---|
| Specify the IP multicast address used for the PTT and paging features | site.cfg > ptt.address |
| Enable Paging mode | site.cfg > ptt.pageMode.enable |
| Specify the display name | site.cfg > ptt.pageMode.displayName |
| Change default settings for Paging mode | site.cfg > ptt.pagemode.* |
| Specify settings for all Page groups | features.cfg > ptt.pageMode.group.* |

Web Configuration Utility

To specify the IP multicast address and port, and available paging groups for Group Paging, navigate to **Settings > Paging/PTT Configuration** and expand **Settings** and **Group Paging Configuration**.

Local Phone User Interface

Specify the IP multicast address and port, and available paging groups for Group Paging from the **Paging/PTT Configuration** menu, accessible from **Menu > Settings > Advanced > Admin Settings**.

Users can access basic Group Paging settings from **Menu > Settings > Basic > Preferences > Paging/PTT Configuration**.



Web Info: Configuring Push-To-Talk and Group Paging

Though the example configurations in this section will get you started, Polycom recommends that you become familiar with the following document before using the PTT or Paging features:

[Feature Profile 62327: Broadcasting Audio Messages with Group Paging and Push-to-Talk](#).

Example PTT/Paging Configuration

The following illustration shows the default PTT and Paging administrator settings in the **site.cfg** template file.

The screenshot shows a configuration editor with a tree view on the left and a corresponding configuration table on the right. The tree view is expanded to show the 'ptt' folder, which contains several sub-folders and parameters. The configuration table lists the following parameters and their values:

| | |
|--------------------------------|-------------|
| ptt.address | 224.0.1.116 |
| ptt.allowOffHookPages | 0 |
| ptt.codec | G.722 |
| ptt.compatibilityMode | 1 |
| ptt.defaultChannel | 1 |
| ptt.displayName | |
| ptt.emergencyChannel | 25 |
| ptt.payloadSize | 20 |
| ptt.port | 5001 |
| ptt.priorityChannel | 24 |
| ptt.pttMode | |
| ptt.pttMode.enable | 0 |
| ptt.callWaiting | |
| ptt.emergencyChannel | |
| ptt.pageMode | |
| ptt.pageMode.allowOffHookPages | 0 |
| ptt.pageMode.codec | G.722 |
| ptt.pageMode.defaultGroup | 1 |
| ptt.pageMode.displayName | |
| ptt.pageMode.emergencyGroup | 25 |
| ptt.pageMode.enable | 0 |
| ptt.pageMode.payloadSize | 20 |
| ptt.pageMode.priorityGroup | 24 |
| ptt.pageMode.transmit | |

Note that you can enter a display name for sent PTT broadcasts in `ptt.displayName` and for sent page announcements in `ptt.pageMode.displayName`.

The two following illustrations show the range of PTT channels and Page groups you can subscribe to.

PTT Mode Channels

You can subscribe to the following PTT channels. Note that channels one and two are enabled by default, and that channels 24 and 25, the priority and emergency channels respectively, are also enabled by default.

The screenshot shows a configuration tree with the following structure:

- pres
 - prov
 - ptt
 - ptt.channel
 - ptt.channel.1.allowTransmit: 1
 - ptt.channel.1.available: 1
 - ptt.channel.1.label
 - ptt.channel.1.subscribed: 1
 - ptt.channel.2.allowTransmit: 1
 - ptt.channel.2.available: 1
 - ptt.channel.2.label
 - ptt.channel.2.subscribed: 0
 - ptt.channel.3.subscribed: 0
 - ptt.channel.4.subscribed: 0
 - ptt.channel.5.subscribed: 0
 - ptt.channel.6.subscribed: 0
 - ptt.channel.7.subscribed: 0
 - ptt.channel.8.subscribed: 0
 - ptt.channel.9.subscribed: 0
 - ptt.channel.10.subscribed: 0
 - ptt.channel.11.subscribed: 0
 - ptt.channel.12.subscribed: 0
 - ptt.channel.13.subscribed: 0
 - ptt.channel.14.subscribed: 0
 - ptt.channel.15.subscribed: 0
 - ptt.channel.16.subscribed: 0
 - ptt.channel.17.subscribed: 0
 - ptt.channel.18.subscribed: 0
 - ptt.channel.19.subscribed: 0
 - ptt.channel.20.subscribed: 0
 - ptt.channel.21.subscribed: 0
 - ptt.channel.22.subscribed: 0
 - ptt.channel.23.subscribed: 0
 - ptt.channel.24.subscribed: 1
 - ptt.channel.25.subscribed: 1
 - ptt.pageMode

Below the tree are two folders: roaming_buddies and roaming_privacy.

Paging Mode Groups

You can subscribe to the following Paging groups. Note that groups one and two are enabled by default, and that groups 24 and 25, the priority and emergency channels respectively, are also enabled by default.

| | |
|------------------------------------|---|
| ptt.channel.24.subscribed | 1 |
| ptt.channel.25.subscribed | 1 |
| ptt.pageMode | |
| ptt.pageMode.group | |
| ptt.pageMode.group.1.allowTransmit | 1 |
| ptt.pageMode.group.1.available | 1 |
| ptt.pageMode.group.1.label | |
| ptt.pageMode.group.1.subscribed | 1 |
| ptt.pageMode.group.2.allowTransmit | 1 |
| ptt.pageMode.group.2.available | 1 |
| ptt.pageMode.group.2.label | |
| ptt.pageMode.group.2.subscribed | 0 |
| ptt.pageMode.group.3.subscribed | 0 |
| ptt.pageMode.group.4.subscribed | 0 |
| ptt.pageMode.group.5.subscribed | 0 |
| ptt.pageMode.group.6.subscribed | 0 |
| ptt.pageMode.group.7.subscribed | 0 |
| ptt.pageMode.group.8.subscribed | 0 |
| ptt.pageMode.group.9.subscribed | 0 |
| ptt.pageMode.group.10.subscribed | 0 |
| ptt.pageMode.group.11.subscribed | 0 |
| ptt.pageMode.group.12.subscribed | 0 |
| ptt.pageMode.group.13.subscribed | 0 |
| ptt.pageMode.group.14.subscribed | 0 |
| ptt.pageMode.group.15.subscribed | 0 |
| ptt.pageMode.group.16.subscribed | 0 |
| ptt.pageMode.group.17.subscribed | 0 |
| ptt.pageMode.group.18.subscribed | 0 |
| ptt.pageMode.group.19.subscribed | 0 |
| ptt.pageMode.group.20.subscribed | 0 |
| ptt.pageMode.group.21.subscribed | 0 |
| ptt.pageMode.group.22.subscribed | 0 |
| ptt.pageMode.group.23.subscribed | 0 |
| ptt.pageMode.group.24.subscribed | 1 |
| ptt.pageMode.group.25.subscribed | 1 |
| roaming_buddies | |
| roaming_privacy | |

Flexible Line Key Assignment

You can give your phone users the ability to assign a line key function to a line key anywhere on the phone's screen. Normally, functions are assigned line keys in succession, the order in which the line key displays on the phone. This feature enables you to break that ordering and assign a line key function to a line key that displays anywhere on the phone's screen. This feature is available on the SoundPoint IP 450, 550, 560, and 650 desktop phones or Expansion Module. Refer to [Table 7-20: Flexible Line Key Assignment](#) for the parameters you will need to configure to set up this feature.

You can apply this feature to any line key function including line appearance, speed dial, busy lamp field (BLF), and presence. Line keys that you configure using this feature will override the

default line key assignments as well as any custom line key configurations you may have made. To use this feature, you will need to specify the function of each line key on the phone. You do this by assigning a category and an index to each line key, both of which are explained in the example configuration.

Specific conditions apply when you assign Busy Lamp Field (BLF) or Presence to line keys. If you are assigning BLF or Presence to a line key, you will need to 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 will be ordered, which BLF contacts will display on line keys 1, 3, and 5. In addition, to assign BLF and Presence to a line key, you will 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.

Table 7-20: Flexible Line Key Assignment

| Central Provisioning Server | template > parameter |
|---|---|
| To enable flexible line key assignment | <code>reg-advanced.cfg > lineKey.reassignment.enabled</code> |
| Specify the line key category | <code>reg-advanced.cfg > lineKey.x.category</code> |
| Specify the line key number (dependent on category) | <code>reg-advanced.cfg > lineKey.x.index</code> |

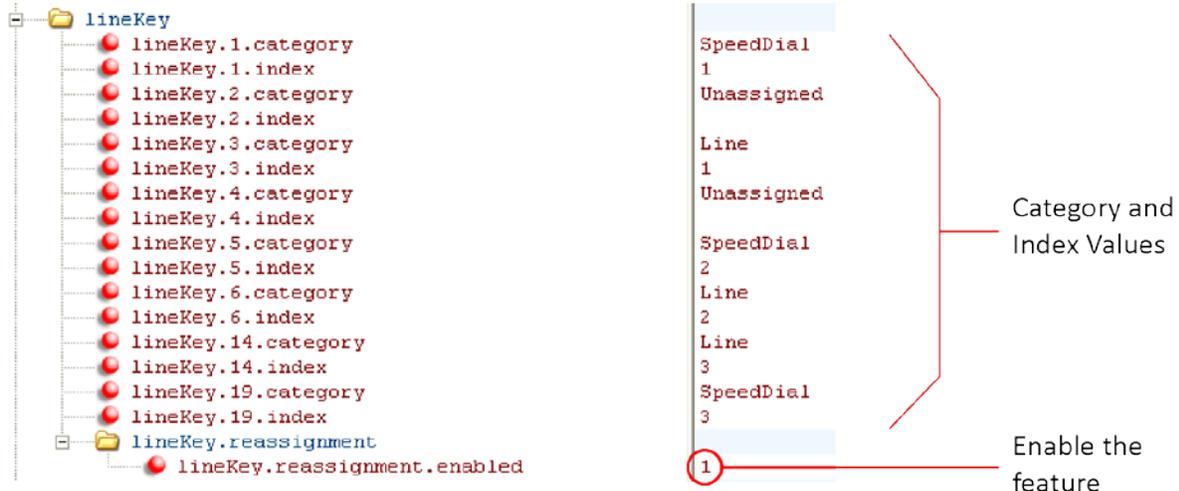
Example Flexible Line Key Assignment Configuration

To enable flexible line key assignment, in the `features.cfg` template, set the `lineKey.reassignment.enabled` parameter to 1. Then assign each line key a category and an index. The category specifies the function of the line key and can include: Unassigned, Line, BLF, SpeedDial, and Presence. Note that the category *Unassigned* will leave that line key blank. The index specifies the order in which the line keys will display on the phone screen. Use [Table 7-21: Assigning Flexible Line Keys](#) to help you assign a category and an index to the line keys on your phone.

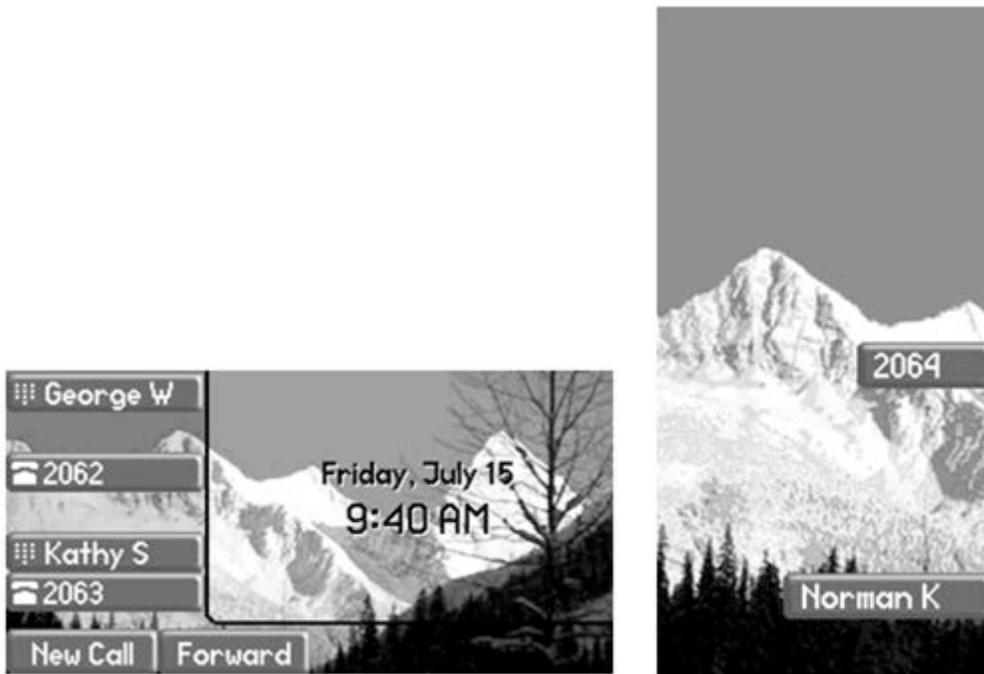
Table 7-21: Assigning Flexible Line Keys

| <i>Assigning a Category and an Index to Line Keys</i> | | | | | |
|---|------------|-----------------------|-----|-----------------------------|----------|
| Category | unassigned | line | BLF | Speed Dial | Presence |
| Index | Null | The line index number | 0 | The speed dial index number | 0 |

The following illustration shows you an example flexible line key assignment configuration in the `features.cfg` template file:



This configuration will display on a SoundPoint IP 650 phone screen as the following:



Configuring Shared Call Appearances

With the shared call appearance feature enabled, an active call displays simultaneously on multiple phones in a group. By default, the answering phone has sole access to the incoming call, called line seize. You can enable another phone in the group the ability to enter a conversation, called a barge in. If the answering phone places the call on hold, that call

becomes available to all phones of that group. The parameters you can configure are listed in [Table 7-22: Configuring Shared Call Appearances](#). All call states of a call —active, inactive, on hold—are displayed on all phones of a group.

This feature is dependent on support from a SIP call server. To enable shared call appearances on your phone, you will need to obtain a shared line address from your SIP service provider. For more details on SIP signaling with shared call appearances, see [Shared Call Appearance Signaling](#).



Tip: Shared Call and Bridged Line Appearances Are Distinct

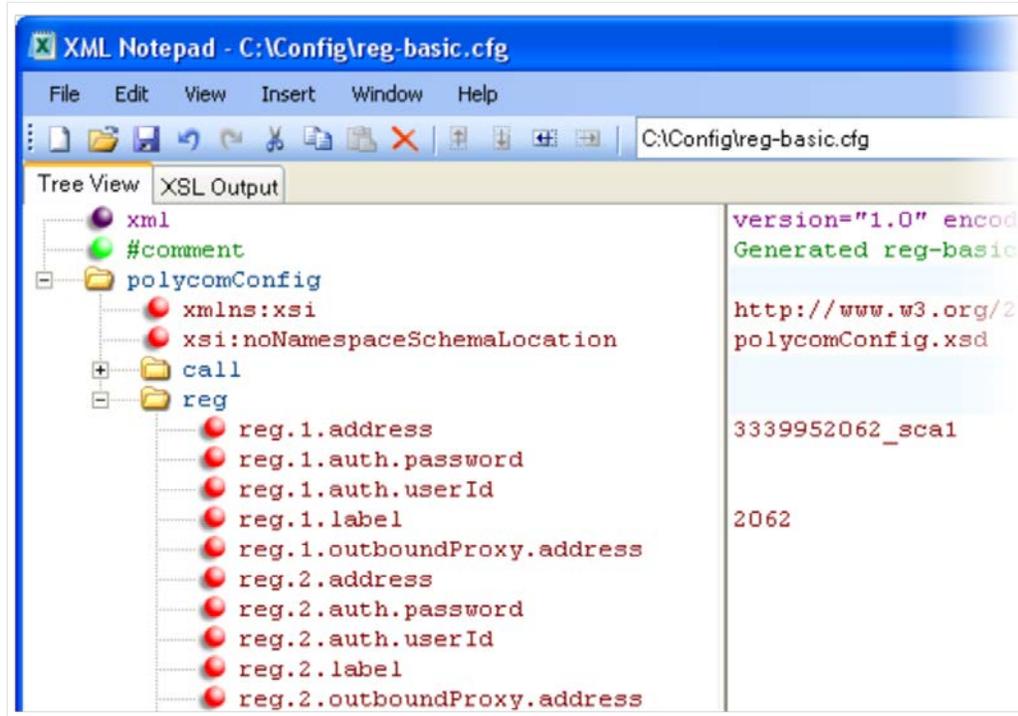
Shared call appearances and bridged line appearances are similar signaling methods that enable more than one phone to share the same line or registration. The method you use varies with the SIP call server you are using.

Table 7-22: Configuring Shared Call Appearances

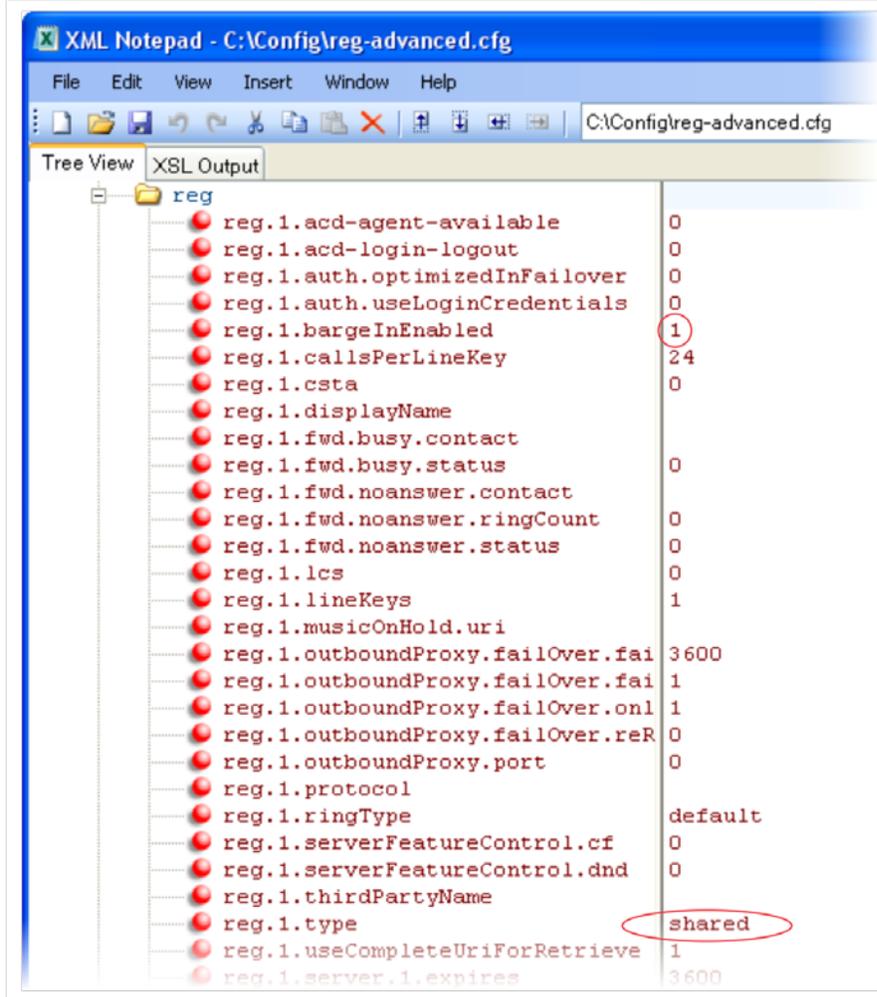
| Central Provisioning Server | template > parameter |
|--|---|
| Specify the shared line address | reg-basic.cfg > reg.x.address |
| Specify the line type as shared | reg-advanced.cfg > reg.x.type |
| To disable call diversion, expose auto-holds, resume with one touch, or play a tone if line-seize fails | sip-interop.cfg > call.shared.* |
| Specify standard or non-standard behavior for processing a line-seize subscription for mutual exclusion | sip-interop.cfg > volpProt.SIP.specialEvent.lineSeize.nonStandard |
| Specify barge-in capabilities and line-seize subscription period if using per-registration servers. A shared line will subscribe to a server providing call state information | reg-advanced.cfg > reg.x.* |
| Specify per-registration whether diversion should be disabled on shared lines | sip-interop.cfg > divert.x.sharedDisabled |
| <hr/> | |
| Web Configuration Utility | |
| To specify the line seize subscription period for SIP Server 1 or Server 2, navigate to Settings > SIP , expand Server 1 or Server 2 , and edit the Line Seize Timeout . | |
| To specify standard or non-standard behavior for processing line-seize subscription for the mutual exclusion feature, navigate to Settings > SIP , expand Local Settings , and enable or disable Non Standard Line Seize . | |
| Specify the per-registration line type (shared) and the line-seize subscription behavior if you are using per-registration server, and whether diversion should be disabled on shared lines by navigating to Settings > Lines . | |
| <hr/> | |
| Local Phone User Interface | |
| To specify the per-registration line type (shared) and shared line address, navigate to Menu > Settings > Advanced > Admin Settings > Line Configuration > Line X > Line Type . | |

Example Configuration

The following illustration shows the address of a registered phone line and the label that displays beside the line key, as specified in the **reg-basic.cfg** template.



If you want to configure this line to be shared, in the **reg-advanced.cfg** template, specify `shared` in `reg.1.type`. All phones that specify `shared` for registration 1 will have shared call appearance enabled for this line. In the following example, the `reg.1.bargeInEnabled` parameter is set to '1' to enable phones of this group to barge in on active calls.



After setting these parameters, activity on line 2062 will display on all phones that configure a shared call appearance for line 2062, as shown in the following illustrations.

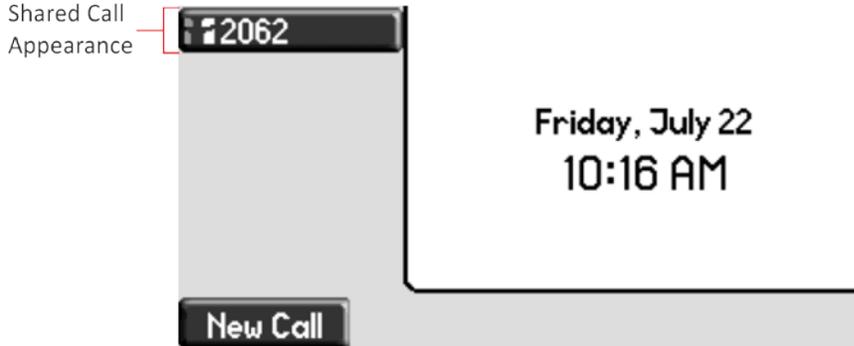
Phone A

In the next illustration, phone A has shared call appearance enabled and is on an active call on line 2062.



Phone B

In the next illustration, phone B has configured a shared call appearance for line 2062. The scrolling phone icon on line key label 2062 shows that this line is in an active call.



Enabling Bridged Line Appearance

Bridged line appearance connects calls and lines to multiple phones. See [Table 7-23: Enabling Bridged Line Appearance](#) for a list of the parameters you can configure. 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—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. For more information, see [Bridged Line Appearance Signaling](#).



Tip: Bridged Line and Shared Call Appearances are Distinct

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 with shared call appearances.

Table 7-23: Enabling Bridged Line Appearance

| Central Provisioning Server | template > parameter |
|---|--|
| Specify whether call diversion should be disabled by default on all shared lines | <code>sip-interop.cfg</code> > call.shared.disableDivert |
| Specify the per-registration line type (private or shared) | <code>reg-advanced.cfg</code> > reg.x.type |
| Specify the shared line third-party name. | <code>reg-advanced.cfg</code> > reg.x.thirdPartyName |
| Specify whether call diversion should be disabled on a specific shared line (overrides default) | <code>reg-advanced.cfg</code> > divert.x.sharedDisabled |

Web Configuration Utility

To specify the line type (private or shared) and the shared line third party name for a specific line, navigate to **Settings > Lines**, choose a line from the left pane, expand **Identification**, and edit **Type** and **Third Party Name**.

To specify whether call diversion should be disabled for a specific shared line, navigate to **Settings > Lines**, choose a line from the left pane, expand **Call Diversion**, and set **Disable Forward for Shared Lines**.

Local Phone User Interface

Specify the line type for each registration and the shared line third party name by navigating to **Menu > Settings > Advanced > Admin Settings > Line Configuration > Line X**. Edit the **Line Type** and the **Third Party Name**.

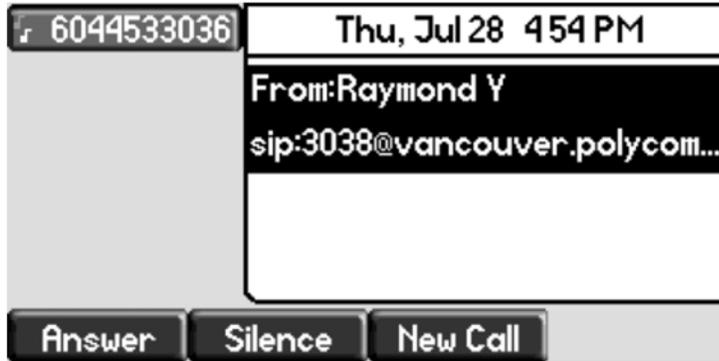
Example Bridged Line Appearance Configuration

To begin using bridged line appearance, you will need to get a registered address dedicated for use with bridged line appearance from your call server provider. This dedicated address must be assigned to a phone line in the `reg.x.address` parameter of the **reg-basic.cfg** template.

Next, in the **reg-advanced.cfg** template, enter the dedicated address in `thirdPartyName` for all phones of the BLA group and set the line type to `shared`. In this example, two or more phones can use the same dedicated address `6044533036` as the BLA address, and the line `type` has been set to `shared` from the default `private`.

| | |
|---|-------------------------|
| <code>reg.1.outboundProxy.port</code> | <code>0</code> |
| <code>reg.1.protocol</code> | <code>default</code> |
| <code>reg.1.ringType</code> | <code>0</code> |
| <code>reg.1.serverFeatureControl.cf</code> | <code>0</code> |
| <code>reg.1.serverFeatureControl.dnd</code> | <code>0</code> |
| <code>reg.1.thirdPartyName</code> | <code>6044533036</code> |
| <code>reg.1.type</code> | <code>shared</code> |
| <code>reg.1.useCompleteUriForRetrieve</code> | <code>1</code> |
| <code>reg.1.server.1.expires</code> | <code>3600</code> |
| <code>reg.1.server.1.expires.lineSeize</code> | <code>30</code> |
| <code>reg.1.server.1.expires.overlap</code> | <code>60</code> |
| <code>reg.1.server.1.lcs</code> | <code>0</code> |
| <code>reg.1.server.1.retryMaxCount</code> | <code>3</code> |
| <code>reg.1.server.1.retryTimeOut</code> | <code>0</code> |
| <code>reg.1.server.1.specialInterop</code> | <code>standard</code> |
| <code>reg.1.server.2.expires</code> | <code>3600</code> |
| <code>reg.1.server.2.expires.lineSeize</code> | <code>30</code> |

In the following example, two phones `6044533036` and `6044533037` are configured with the `3036` BLA address. There is an incoming call to `6044533036` from `3038` that causes `3036` and `3037` phones to show the incoming call, as shown next.



BLA with Line
6044533036



Using Busy Lamp Field

The busy lamp field (BLF) feature enables users to monitor the status of lines on remote phones, display remote party information, and answer incoming calls to remote phones (called directed call pickup). The BLF feature must be supported by a call server and the specific functions will vary with the call server you use. You may need to consult your SIP server partner or Polycom channel partner to find out how to configure BLF.

[Table 7-24: Busy Lamp Field](#) lists the parameters you may need to set. You can set up multiple BLF lines and monitor remote phones in active, ringing, and idle state. When BLF is enabled and you are monitoring a remote user, a BLF line key icon will display on the phone's screen. You can configure the line key label, and how call appearances and caller ID information are displayed. As of SIP 3.2.0, you can configure one-touch call park and retrieve and one-touch directed call pickup. Specifying the type of monitored resource as normal or automata changes the default actions of key presses. As the resource type, enter `normal` if the monitored resource type is a phone and `automata` if the monitored resource type is, for example, a call orbit. If you select `normal`, pressing the BLF line key will place an active call on hold before dialing the selected BLF phone. If you select `automata`, pressing the BLF line key will immediately transfer active calls to that resource. To learn how to configure a park orbit and for examples, see [Configuring Enhanced Feature Keys](#).

Note that how you manage calls on BLF lines depends on the state of your phone — whether it is in the idle, active, or alerting state.



Web Info: Managing Monitored Lines

For information on how to manage calls to monitored phones, see the section Handling Remote Calls on Attendant Phones in [Technical Bulletin 62475: Using Statically Configured Busy Lamp Field with Polycom® SoundPoint IP Phones](#).



Note: VVX Phones Do Not Display All BLF Information

Note that VVX phones do not display the call state of monitored phones or the caller ID of incoming calls to a monitored phone.

As of the SIP 3.1.0 release, the BLF feature was updated in the following ways:

- The phone will give a visual and audible indication when monitored BLF lines have incoming calls.
- The phone will display the caller ID of incoming calls to a remote monitored phones. BLF lines display a Pickup soft key that you can press to answer incoming calls to that monitored resource.

As of the SIP 3.2 release, the BLF feature was updated in the following ways:

- You can create a list of monitored parties to a maximum of 47 and configure the line key labels.
- You can configure key functions.
- You can disable spontaneous call appearances from incoming calls on monitored lines.

The following call servers are known to support this feature:

- Back to Back2 User Agent (B2BUA) Architecture
 - Metaswitch Metasphere Call Feature Server (CFS)
 - Asterisk® v1.6 or later
 - BroadSoft® BroadWorks
- Proxy Architecture
 - Avaya® SipX Enterprise Communications Server (ECS)
 - eZuce openUC™

These proxy architectures may support the full range of statically configured BLF features. However, they do not provide configuration control through their Web management console.

The following call servers may support this feature, depending on the call server software variation and deployment:

- Proxy Architecture
 - OpenSIPS (formerly OpenSER)
 - ReSIPProcate

These proxy architectures or any other proxy server that allows the phone end-to-end communications with the monitored phone should be supported. However, these solutions have not been specifically tested by Polycom nor does Polycom guarantee their full interoperability.



Tip: Polycom Phones Compatible with BLF

This feature is available on SoundPoint IP 450, 550, 560, 650 phones, and VVX phones running software SIP 2.1 to UC Software 3.1.0. Other phone models may be monitored, but cannot be configured to monitor other phones.



Note: BLF Not Compatible with Microsoft Live Communications Server 2005

Polycom recommends that the BLF not be used in conjunction with the Microsoft Live Communications Server 2005 feature. For more information, see Microsoft Live Communications Server 2005 Integration.



Settings: Use BLF With TCPpreferred Transport

Use this feature with TCPpreferred transport (see [<server/>](#)). You can also use UDP transport on SoundPoint IP 650 phones.

Table 7-24: Busy Lamp Field

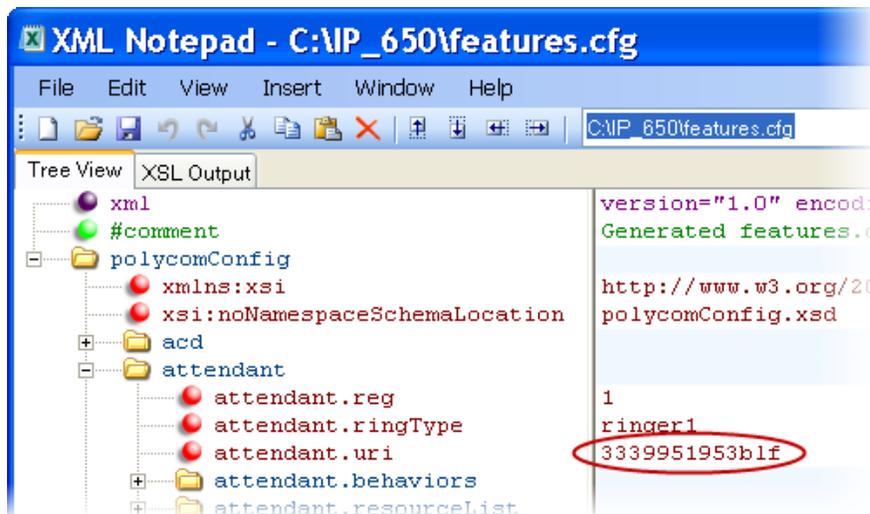
| Central Provisioning Server | template > parameter |
|---|--|
| Specify an index number for the BLF resource | features.cfg > attendant.reg |
| Specify the ringtone to play when a BLF dialog is in the offering state | features.cfg > attendant.ringType |
| Specify the SIP URI of the call server resource list | features.cfg > attendant.uri |
| Specify how call appearances and remote party caller ID display on the attendant phone | features.cfg > attendant.behaviours.display.* |
| Specify the address of the monitored resource, a label for the resource, and the type of resource | features.cfg > attendant.resourceList.* |

Example BLF Configuration

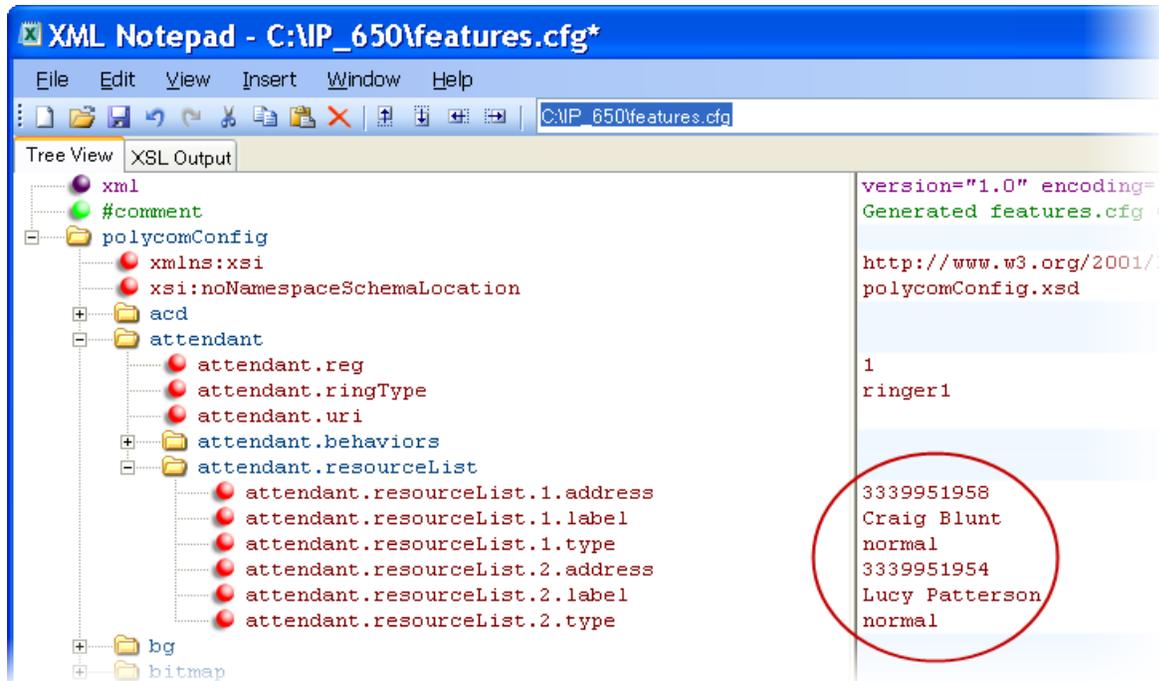
Typically, call servers support one of two methods of BLF configuration. Using the first method, you subscribe to a BLF resource list that is set up on your call server. Using the second method, you enter BLF resources to a configuration file and the call server directs the requests to those

BLF resources. If you are unsure which method to use, consult your SIP server partner or Polycom Channel partner. This section shows you how to set up BLF using both methods.

To subscribe to a BLF list on a call server, you will need to access the call server and set up a list of monitored resources. The call server will provide 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` field located in the `features.cfg` template file, as shown next.



To specify BLF resources in the configuration file, open the `features.cfg` template file and enter the address (phone number) of the BLF resource you want to monitor, the label that will display beside the line key on the phone, and the type of resource you are monitoring. Your call server must support static BLF in order to configure BLF using the static method. In the following example, the phone is monitoring *Craig Blunt* and *Lucy Patterson*:



Both configuration methods result in the following BLF contacts – called BLF resources – beside line keys on the phone:



The following table illustrates the BLF key icons.

Table 7-25: BLF Line Key Icons

| States | Line Icons |
|---------------------------|---|
| Line monitoring is active |  |
| Monitored line is busy |  |
| Monitored line is ringing |  |



Web Info: Using the BLF Feature

For details on using the BLF feature, see [Quick Tip 37381: Understanding Enhanced BLF on SoundPoint IP Phones](#).

Enabling Voicemail Integration

The phone is compatible with voicemail servers. You can configure each phone or line registration per phone to subscribe with a SIP URL to a voicemail server contact. You can also configure the phone to access voicemail with a single key, for example, the **Messages** key on the SoundPoint IP 450, 550, 560, and 650 phones, the **MSG** key on the VVX 1500 phone, and the **Messages** icon on the VVX 500 phone and SpectraLink handset's Home screen. When you access the voicemail server, the phone gives a visual and audio alert; you can also configure a message waiting alert to indicate that you have unread voicemail messages [Table 7-26: Voicemail Integration](#) shows you the parameters you can configure.

Table 7-26: Voicemail Integration

| Central Provisioning Server | template > parameter |
|---|--|
| To turn one-touch Voicemail on or off
..... | sip-interop.cfg > up.oneTouchVoiceMail |
| Specify the URI of the message center server | sip-interop.cfg > msg.mwi.x.subscribe |
| Set the mode of message retrieval | sip-basic.cfg > msg.mwi.x.callBackMode |
| Specify a contact number for the phone to call to retrieve messages, <code>callBackMode</code> must be set to Contact
..... | sip-interop.cfg > msg.mwi.x.callBack |
| Specify if message waiting notifications should display or not | site.cfg > up.mwiVisible |
| <hr/> | |
| Web Configuration Utility | |
| To turn One Touch Voicemail on or off, navigate to Preferences > Additional Preferences , expand User Preferences , and set One Touch Voicemail . | |
| To specify the message center settings for a specific line, navigate to Settings > Lines , select a line from the left pane, and expand Message Center . | |

Example Voicemail Configuration

The following illustration shows you how to enable one-touch access to the voicemail server. In the next illustration, line 2 is configured to subscribe to the voicemail server at *voicemail.polycom.com*.

| Configuration Path | Value |
|----------------------------------|-----------------------|
| msg.bypassInstantMessage | 0 |
| msg.mwi.1.callBack | |
| msg.mwi.1.subscribe | |
| msg.mwi.2.callBack | |
| msg.mwi.2.subscribe | voicemail.polycom.com |
| up.callTypePromptPref | 1 |
| up.enableCallTypePrompt | 1 |
| up.idleTimeout | 40 |
| up.lineKeyCallTerminate | 0 |
| up.lineKeyCallTerminate.SPIP235T | 1 |
| up.numberFirstCID | 0 |
| up.oneTouchVoiceMail | 1 |
| up.prioritizeBackgroundMenuItem | |
| up.screenCapture | |

The following illustration shows that, in the **sip-basic.cfg** template, the default `callBackMode` setting for line 2 is set to `registration`. The phone will use the address assigned to line 2 to subscribe to the voicemail server you entered in `msg.mwi.2.subscribe`.

```

version="1.0" encoding="UTF-8"
Generated sip-basic.cfg

http://www.w3.org/2001/XMLSchema-instance
polycomConfig.xsd

registration
  
```

Once this is enabled in the **sip-interop.cfg** template, on the phone, press the **Messages** key and select **Message Center** to access your voicemail.

Enabling Multiple Registrations

Polycom phones can have multiple registrations; each registration requires an address, or phone number. [Table 7-27: Enabling Multiple Registrations](#) explains the registration parameters and options. The IP 321, 331, and 335 phones support a maximum of two registrations, the IP 450 phones support up to three, the IP 550 and 560 phones support up to four, and the IP 650, VVX 500 and 1500 phones and SpectraLink handsets support up to six. Up to three SoundPoint IP Expansion Modules can be added to a single host SoundPoint IP 650 phone to increase the total number of registrations to 34. The SoundStation IP 5000 and 6000 each support a single registration.

Each registration can be mapped to one or more line keys. Note that a line key can be used for only one registration. The user can select which registration to use for outgoing calls or which to use when initiating new instant message dialogs. Note that this feature is one of several features associated with *Flexible Call Appearances*. For definitions of all features associated with Flexible Call Appearances, see [Table 7-4: Flexible Call Appearances](#).

Table 7-27: Enabling Multiple Registrations

| Central Provisioning Server | template > parameter |
|---|----------------------|
| Specify the local SIP signaling port and several optional SIP servers to register to. For each server specify the registration period and the signaling failure behavior sip-interop.cfg
> volpProt.SIP.* and volpProt.server.x.* | |
| Specify a display name, a SIP address, an optional display label, an authentication user ID and password, the number of line keys to use, and an optional array of registration servers. The authentication user ID and password are optional and for security reasons can be omitted from the configuration files. The local flash parameters will be used instead. The optional array of servers and their parameters will override the servers specified in <volpProt.server/> if non-Null reg-basic.cfg , reg-advanced.cfg > reg.x.* | |

Web Configuration Utility

Specify the local SIP signaling port and several optional SIP servers to register to.

Specify a display name, a SIP address, an optional display label, an authentication user ID and password, the number of line keys to use, and an optional array of registration servers. The authentication user ID and password are optional and for security reasons can be omitted from the configuration files. The local flash parameters will be used instead. The optional array of servers will override the servers specified in <server/> in non-Null.

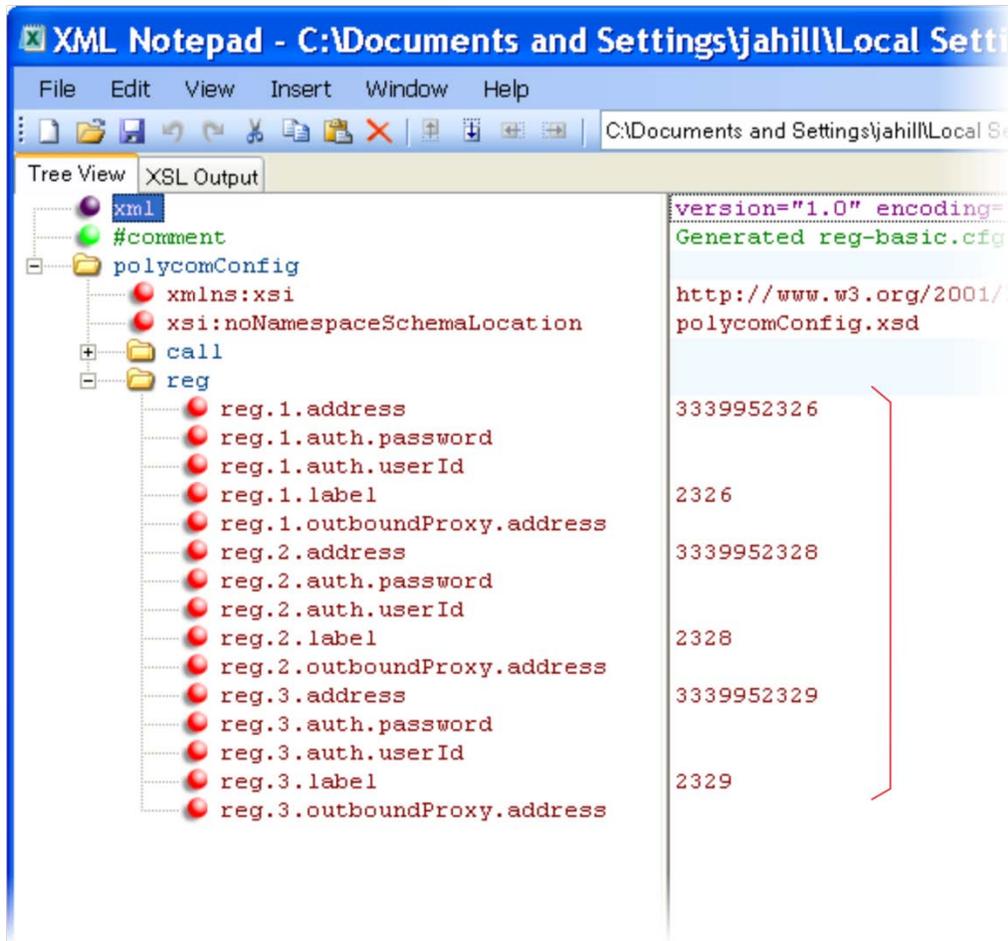
Configure multiple registrations by navigating to **Settings > Lines**.

Local Phone User Interface

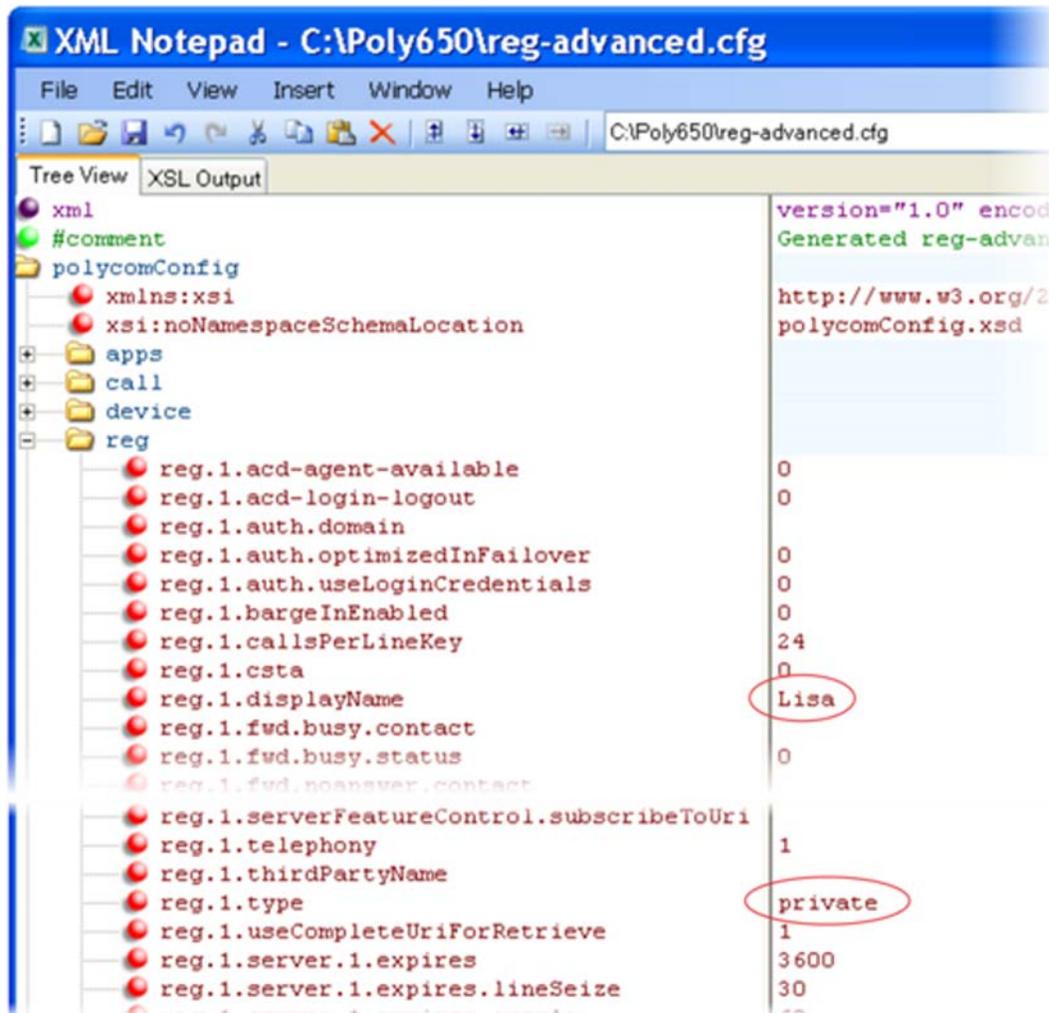
Use the **Call Server Configuration** and **Line Configuration** menu to specify the local SIP signaling port, a default SIP server to register to, and registration information for up to twelve registrations (depending on the phone model). These configuration menus contain a sub-set of all the parameters available in the configuration files.

Example Multiple Registration Configuration

In the next illustration, in the **reg-basic.cfg** template, multiple line registrations and a label for each registration has been enabled for lines 1, 2, and 3.



In the **reg-advanced.cfg** template shown next, when you make a call using line 1, the name you enter in `reg.1.displayName` will display as your caller ID, in this case *Lisa*. The parameter `reg.x.type` is left in the default `private`, which indicates that the registration will use standard call signaling.



This configuration will result in the following registrations on a SoundPoint IP 650 phone:



Using Hoteling

The Hoteling feature enables users to use any available shared phone by logging in to a guest profile. After logging in, users have access to their own guest profile and settings on the shared phone. This feature is available on Polycom SoundPoint IP 450, 550, 560, and 650 phones configured with the BroadSoft BroadWorks R17 platform and running UC Software 4.0.2 or later.



Web Info: Using the Hoteling Feature

For details on configuring the Hoteling feature, [Using Hoteling on Polycom Phones \(Feature Profile 76413\)](#).

You can use Hoteling in conjunction with the Feature-Synchronized Automatic Call Distribution (ACD) feature (ACD). For information, see [Configuring Feature-Synchronized Automatic Call Distribution](#).

Table 7-28: Using Hoteling

| Central Provisioning Server | template > parameter |
|---------------------------------------|---|
| Enable or disable Hoteling | features.cfg > feature.hoteling.enabled |
| Choose a line registration index..... | features.cfg > hoteling.reg |

Example Hoteling Configuration

This example configuration shows the hoteling feature enabled and uses registration line 1. In the **features.cfg** template, the `feature.hoteling.enabled` parameter is set to 1 to enable.



The hoteling feature is applied to phone line 1.



When hoteling is enabled, the line 1 index key 2326 has hoteling enabled and the **GuestIn** soft key displays.



Configuring SIP-B Automatic Call Distribution

As of SIP 3.1.2, you can use your SoundPoint IP phones in a call center agent/supervisor role on a supported call server. Automatic call distribution (ACD) enables organizations that handle a large number of incoming phone calls to use SoundPoint IP phones in a call center role. SIP-B ACD parameters are listed in [Table 7-28: Configuring SIP-B Automatic Call Distribution](#) and [Table 7-29: ACD Agent Availability](#). SIP-B is a basic version of the ACD feature. If you are using Feature Synchronized ACD, see [Configuring Feature Synchronized Automatic Call Distribution](#).

Only the SoundPoint IP phones support Automatic Call Distribution.

The SoundPoint IP phones support SIP-B ACD login and logout. This feature depends on support from a SIP server.

Table 7-29: Configuring SIP-B Automatic Call Distribution

| Central Provisioning Server | template > parameter |
|--|---|
| To turn Automatic Call Distribution on or off | features.cfg > feature.acdLoginLogout.enabled |
| To enable or disable Automatic Call Distribution for a specific registration | reg-advanced.cfg > reg.x.acd-login-logout |
| To enable or disable Feature Synchronized ACD | sip-interop.cfg > volpProt.SIP.acd.signalingMethod |

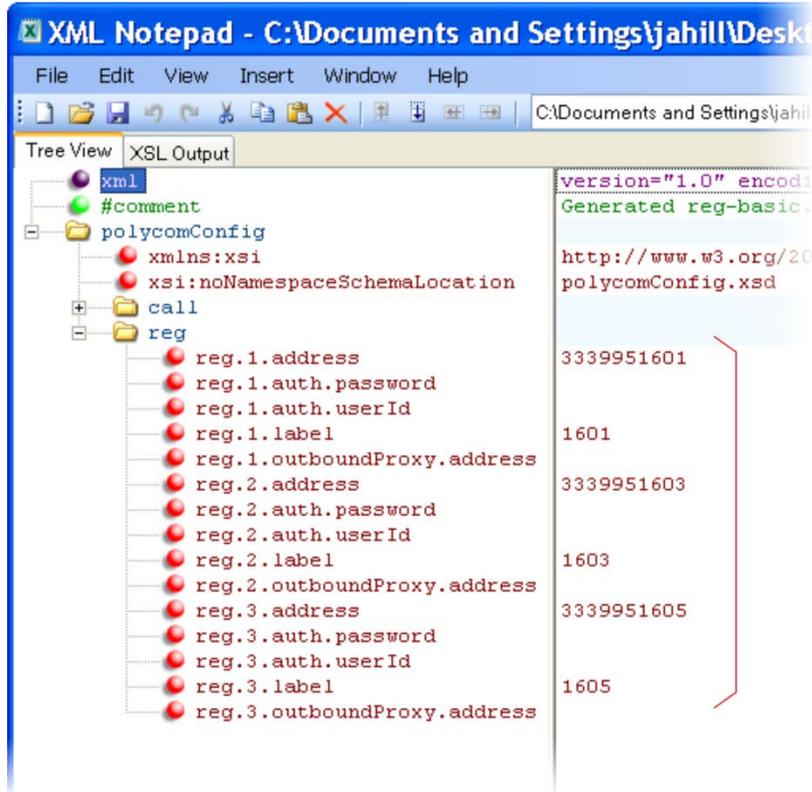
The SoundPoint IP phones also support ACD agent availability. This feature depends on support from a SIP server.

Table 7-30: ACD Agent Availability

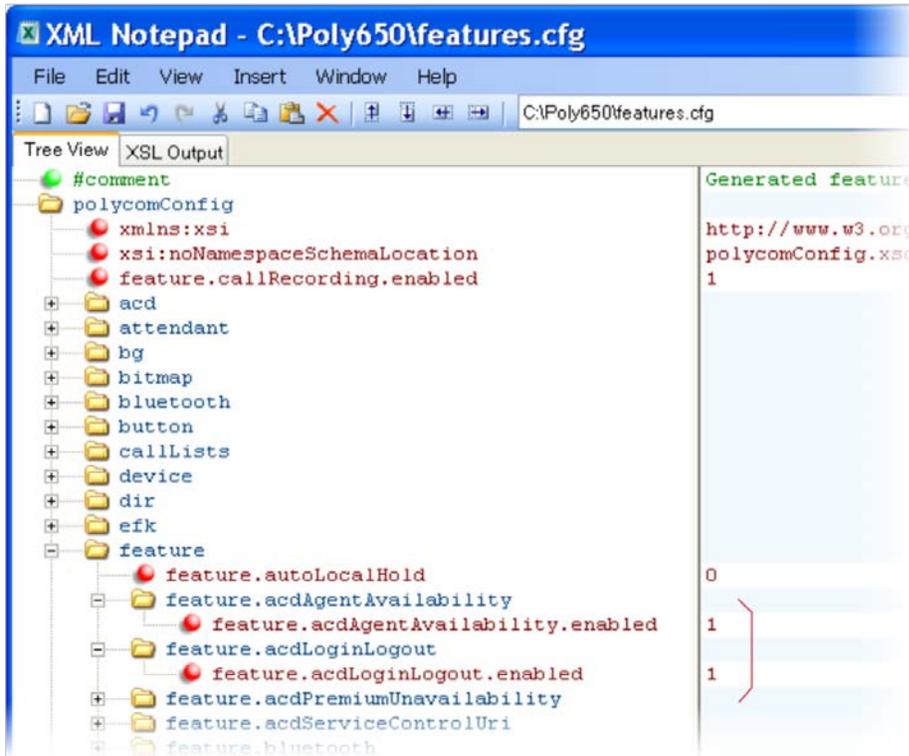
| Central Provisioning Server | template > parameter |
|---|---|
| To turn ACD Agent Availability on or off | features.cfg > feature.acdAgentAvailable.enabled |
| To enable or disable ACD Agent Availability feature for a specific registration | reg-advanced.cfg > reg.x.acd-agent-available |

Example SIP-B Automatic Call Distribution Configuration

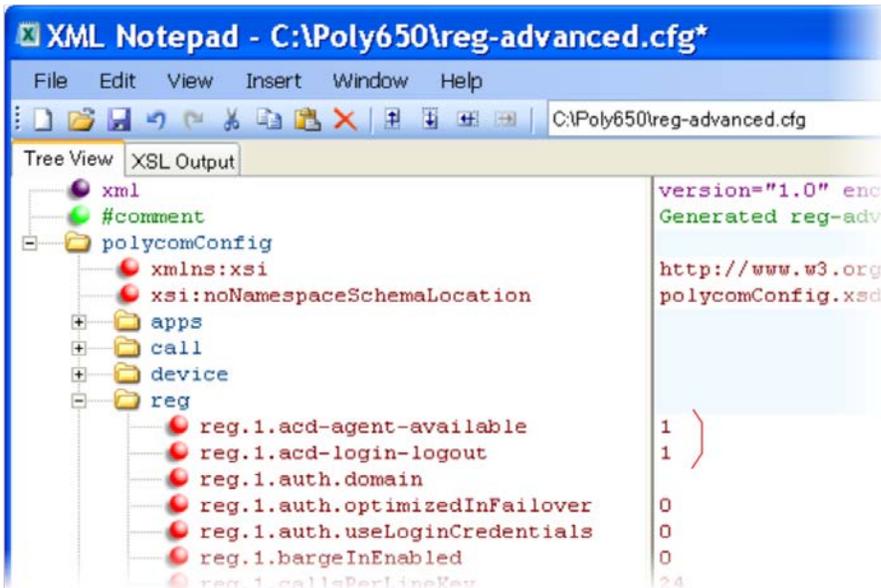
In the following illustration, in the **reg-basic.cfg** template file, three line registrations and labels have been set up.



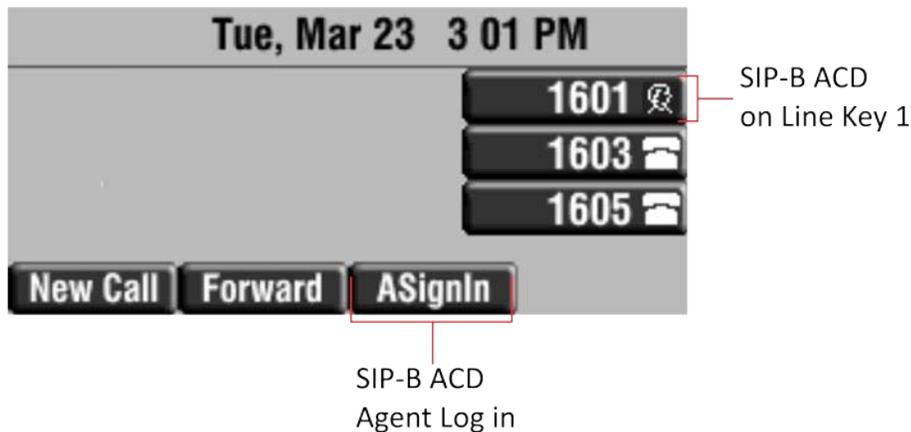
In this example, SIP-B ACD is enabled in **features.cfg** using the parameters `feature.acdAgentAvailability.enabled` and `feature.acdLoginLogout.enabled`, as shown next.



You will also need to enable SIP-B ACD in the **reg-advanced.cfg** template file. The next illustration shows the two parameters you need to enable to display the ACD soft keys on the phone screen.



Once SIP-B ACD is enabled, the following soft keys will display on the phone.



The ACD agent 1601 displays on phone line 1 and the agent can log in and out of the ACD feature.

Configuring Feature-Synchronized Automatic Call Distribution (ACD)

As of SIP 3.1.2, you can use your SoundPoint IP phones in a call center agent/supervisor role on a supported call server. Feature-Synchronized ACD is distinct from and provides more advanced ACD functions than the [Configuring SIP-B Automatic Call Distribution](#) feature.

Feature-synchronized Automatic Call Distribution (ACD) enables organizations that handle a large number of incoming phone calls to use SoundPoint IP phones in a call center role.

Feature-synchronized ACD is available as a standard or a premium service. The premium ACD service has been enhanced in two ways: *Hoteling* and *Queue Status Notification*. Hoteling enables agents to use their agent credentials to log in to any available phone. If you want to use the Hoteling feature with Feature-Synchronized ACD, see [Using Hoteling](#). Queue Status Notification enables agents to view the queue status of a call center so that agents can adjust their call response.



Web Info: Further Information on ACD Enhancements

For more information on standard and premium ACD as well as the Hoteling and Queue Status Notification enhancements, see *Using Premium Automatic Call Distribution for Call Centers (Feature Profile 76179)* on [Polycom Profiled UC Software Features](#).

See [Table 7-30: Configuring Feature Synchronized Automatic Call Distribution](#) for parameters you can configure. When standard functions are enabled, the phone will indicate it is in the ACD Call Center Agent state. Phone users can sign in and sign out of the ACD state as a call center agent using soft keys or the phone's menu. When ACD is enabled and a user is signed in as an agent, the phone can display the current state of the agent, for example, whether the agent is available or unavailable to take new calls.

The capabilities of this feature vary with the SIP call server. Please 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*.

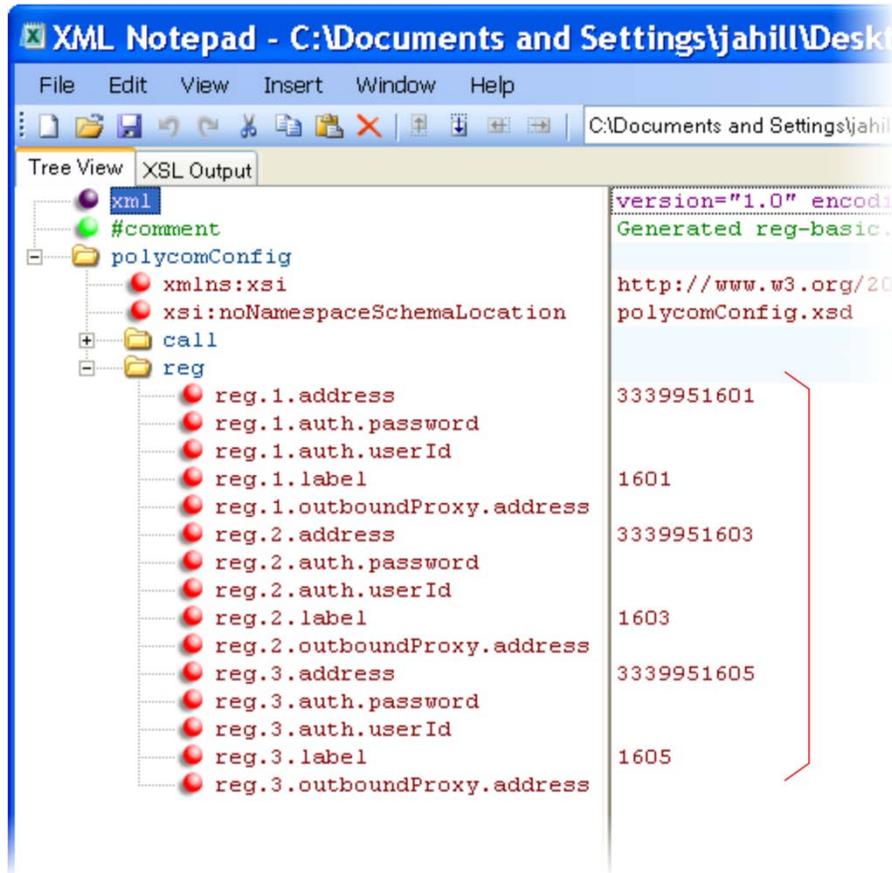
The following phones support the Feature-synchronized ACD feature: SoundPoint IP 321/331/335, 450, 550, 560, and 650 phones, and the VVX 500. Note that the Hoteling and Queue Status Notification enhancements are not available on the VVX 500.

Table 7-31: Configuring Feature Synchronized Automatic Call Distribution

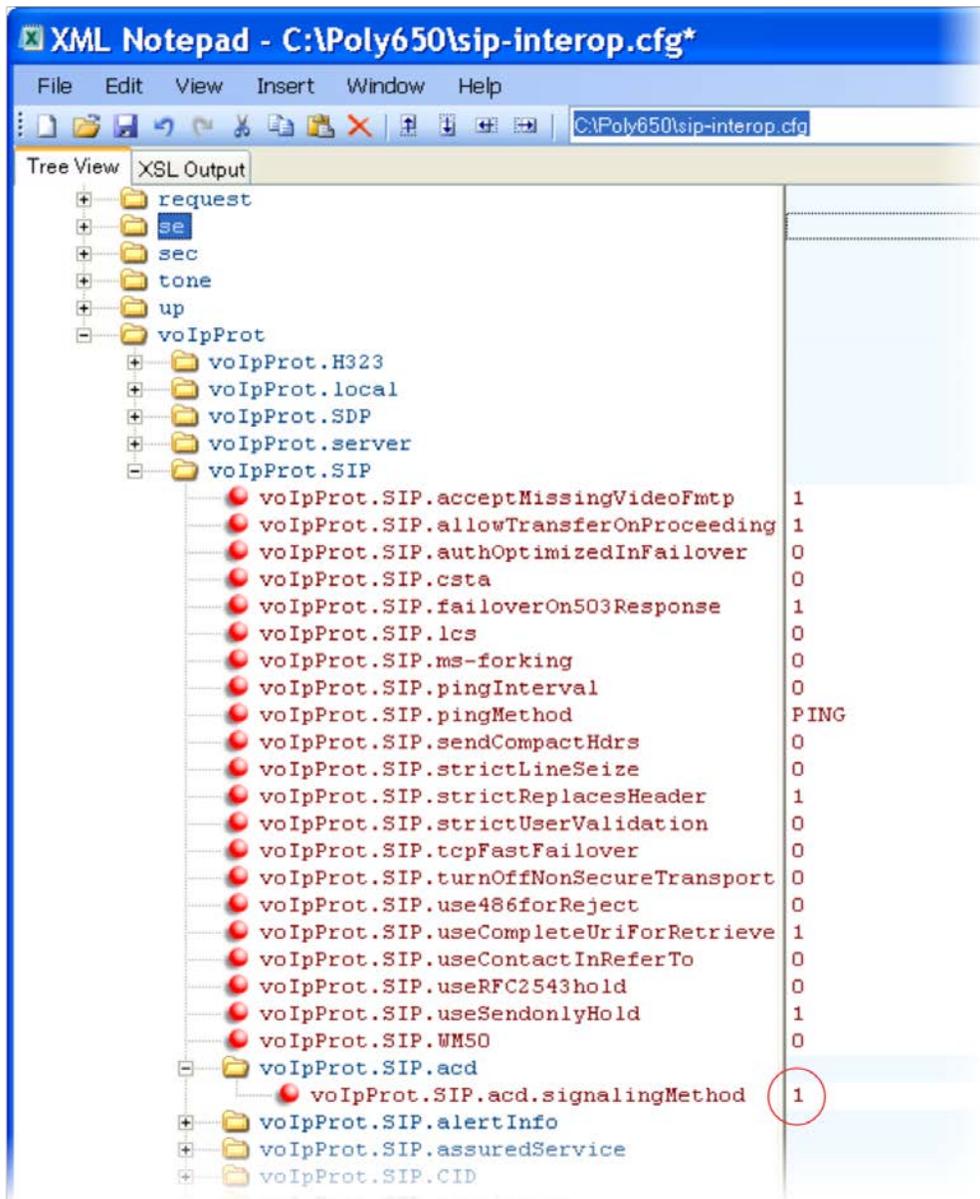
| Central Provisioning Server | template > parameter |
|---|--|
| To turn Feature Synchronized ACD on or off | features.cfg > feature.acdLoginLogout.enabled |
| To turn ACD Agent Availability on or off | features.cfg > feature.acdAgentAvailable.enabled |
| To turn Premium Feature Synchronized ACD on or off
..... | features.cfg > feature.acdPremiumUnavailability.enabled |
| To turn Feature Synchronized ACD Control URI on or off | features.cfg
> feature.acdServiceControlUri.enabled |
| To set the registration to be used for Feature Synchronized ACD and the users' sign-in state
..... | features.cfg > acd.* |
| To enable or disable Feature Synchronized ACD . | sip-interop.cfg > volpProt.SIP.acd.signalingMethod |

Example Feature Synchronized ACD Configuration

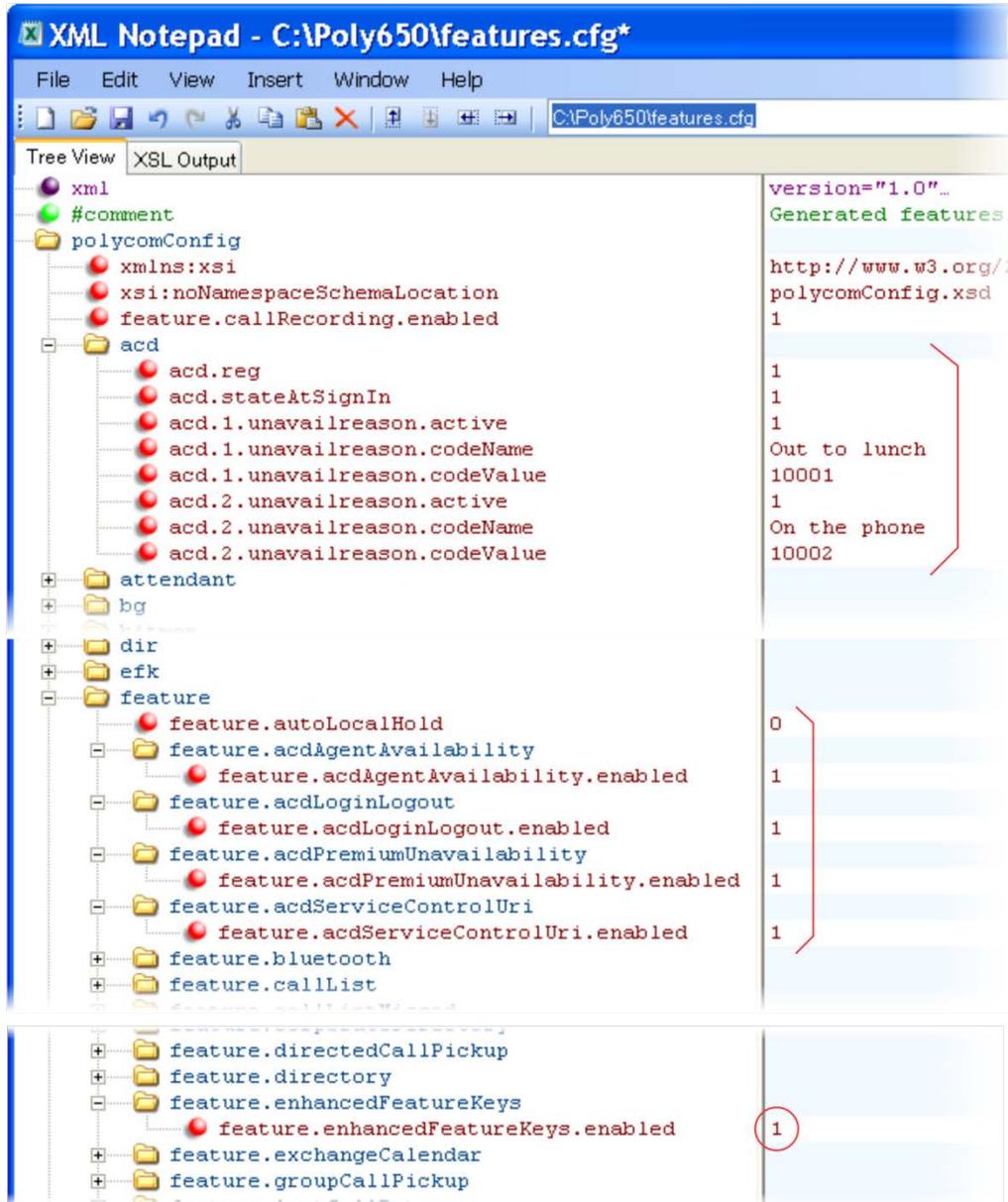
In the following illustration, in the **reg-basic.cfg** template file, three line registrations and labels have been set up.



To enable Feature Synchronized ACD for these registrations, in the **sip-interop.cfg** template file, set `voIpProt.SIP.acd.signalingMethod` to 1, as shown next.



As shown next, you will need to enable the `feature.enhancedFeaturekeys.enabled` parameter, four parameters in `feature.acd*`, and the `acd.reg` and `acd.stateAtSignIn` parameters. If you want to use reason codes, enable `acd.unavailreason.active` and enter the reason codes in the `acd.x.unavailreason.codeName` and `acd.x.unavailreason.codeValue` parameters. You can define up to 100 reason codes. In the following example, two reason codes have been enabled and set, *Out to lunch* and *On the phone*.



This configuration results in the following screens on the SoundPoint IP 450 phone. When the agent presses the **Unavailable** soft key, the reason codes you entered will display.



The ACD agent 1601 displays on phone line 1 and the agent can log in and out of the ACD feature.



Web Info: Configuration Details for Feature Synchronized ACD

For details on how to configure SoundPoint IP phones for Feature Synchronized ACD, see [Technical Bulletin 57216: Using Feature Synchronized Automatic Call Distribution with Polycom SoundPoint IP Phones](#).

Setting Up Server Redundancy

Server redundancy is often required in VoIP deployments to ensure continuity of phone service if, for example, where the call server needs to be taken offline for maintenance, the server fails, or the connection between the phone and the server fails. [Table 7-31: Setting Up Server Redundancy](#) points to several parameters you can configure.

Two types of redundancy are possible:

- **Failover**—In this mode, full phone system functionality is preserved by having a second call server of equivalent capability take over from the server that went down/off-line. Use this mode of operation with DNS mechanisms or ‘IP Address Moving’ from the primary to the back-up server.

- **Fallback**—In this mode, a second call server of lesser capability (router or gateway device) takes over call control to provide basic calling capability without some of the richer features offered by the primary call server (for example, shared lines, presence, and Message Waiting Indicator). Polycom phones support configuration of multiple servers per SIP registration for this purpose.

In some cases, a combination of the two may be deployed. Consult your SIP server provider for recommended methods of configuring phones and servers for failover configuration.



Note: Compatibility with Microsoft® Lync

The concurrent failover/fallback feature is not compatible with Microsoft Lync.



Caution: Old Failover Behavior Is Not Supported

Prior to SIP 2.1, the `reg.x.server.y` parameters in `<reg/>` could be used for failover configuration. The older behavior is no longer supported. Customers that are using the `reg.x.server.y.*` configuration parameters where $y \geq 2$ should take care to ensure that their current deployments are not adversely affected. For example, the phone will only support advanced SIP features such as shared lines, missed calls, and presence with the primary server ($y=1$).

Table 7-32: Setting Up Server Redundancy

| Central Provisioning Server | template > parameter |
|--|---|
| Specify server redundancy options including fallback mode, fallback timeout, and failover registration behaviour | <code>sip-interop.cfg</code> > <code>volpProt.server.x.failOver.*</code> |
| Specify which server to contact if failover occurs ... | <code>reg-advanced.cfg</code> > <code>reg.x.auth.optimizedInFailover</code> |
| Override the default server redundancy options for a specific registration | <code>reg-advanced.cfg</code> > <code>reg.x.outboundProxy.failOver.*</code> |



Web Info: Failover Configuration Details

For more information, see [Technical Bulletin 5844: SIP Server Fallback Enhancements on Polycom Phones](#) and [Technical Bulletin 66546: Using Optional Geographical Server Redundancy Failover Behaviors](#).

DNS SIP Server Name Resolution

If a DNS name is given for a proxy/registrar address, the IP address(es) associated with that name will be discovered as specified in RFC 3263. If a port is given, the only lookup will be an A record. If no port is given, NAPTR and SRV records will be 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, 5060 will be used. If the registration type is Transport Layer Security (TLS), 5061 will be used as the port number. See [RFC 3263](#) for an example.



Caution: No DNS Resolution Will Cause Failover

Failure to resolve a DNS name is treated as signaling failure that will cause a failover.

Behavior When the Primary Server Connection Fails

For Outgoing Calls (INVITE Fallback)

When the user initiates a call, the phone will go through the following steps to connect the call:

- 1 The phone will try to call the working server.
- 2 If the working server does not respond correctly to the INVITE, the phone will try and make 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 will try 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 will fail.

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. For more information, see [<server/>](#) and [<reg/>](#).



Caution: Use Long TTLs to Avoid DNS Timeout Delays

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 will attempt to contact the DNS server to resolve the address of all servers in its list *before* initiating a call. These attempts will 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. Polycom recommends deploying an on-site DNS server as part of the redundancy solution.

Phone Configuration

The phones at the customer site are configured as follows:

- Server 1 (the primary server) will be configured with the address of the service provider call server. The IP address of the server(s) will be provided by the DNS server, for example: `reg.1.server.1.address=voipserver.serviceprovider.com` .
- Server 2 (the fallback server) will be 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` .



Note: Caution When Using Multiple Servers Per Registration

It is possible to configure the phone for more than two servers per registration, but you need to exercise caution when doing this to ensure that the phone and network load generated by registration refresh of multiple registrations does not become excessive. This would be of particular concern if a phone had multiple registrations with multiple servers per registration and it is expected that some of these servers will be unavailable.

Phone Operation for Registration

After the phone has booted up, it will register to all the servers that are configured.

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 will be established only with Server 1.

Upon the registration timer expiry of each server registration, the phone will attempt to re-register. If this is unsuccessful, normal SIP re-registration behavior (typically at intervals of 30 to 60 seconds) will proceed and continue until the registration is successful (for example, when the Internet link is once again operational). While the primary server registration is unavailable, the next highest priority server in the list will serve as the working server. As soon as the primary server registration succeeds, it will return to being the working server.

**Note: Failover to Servers that are Not Registered**

If `reg.x.server.y.register` is set to 0, the phone will not register to that server. However, the INVITE will fail 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. SoundPoint IP phones can only be configured with one OutBoundProxy per registration and all traffic for that registration will be routed through this proxy for all servers attached to that registration. If Server 2 is not accessible through the configured proxy, call signaling with Server 2 will fail.
- Avoid using too many servers as part of the redundancy configuration as each registration will generate more traffic.
- Educate users as to the features that will not be available when in fallback operating mode.

**Note: Compatibility with Microsoft® Lync**

The concurrent/registration failover/fallback feature is not compatible with Microsoft® Lync.

Using the Presence Feature

The presence feature enables you to monitor the status of other remote users and phones. By adding remote users to your Buddy List, you can monitor changes in the status of remote users in real time or you can monitor remote users as speed-dial contacts. You can also manually specify your status in order to override or mask automatic status updates to others and you can receive notifications when the status of your a remote line changes. [Table 7-32: Using the Presence Feature](#) lists the parameters you can configure. Note that other phone users can block you from monitoring their phones. The SpectraLink handsets support only the Microsoft® Live Communications Server 2005 and Microsoft Office Communications Server 2007 R2 versions of Presence (see [Setting Up Microsoft Live Communications Server 2005 Integration](#) and [Setting Up Microsoft Office Communications Server 2007 R2 Integration](#)).

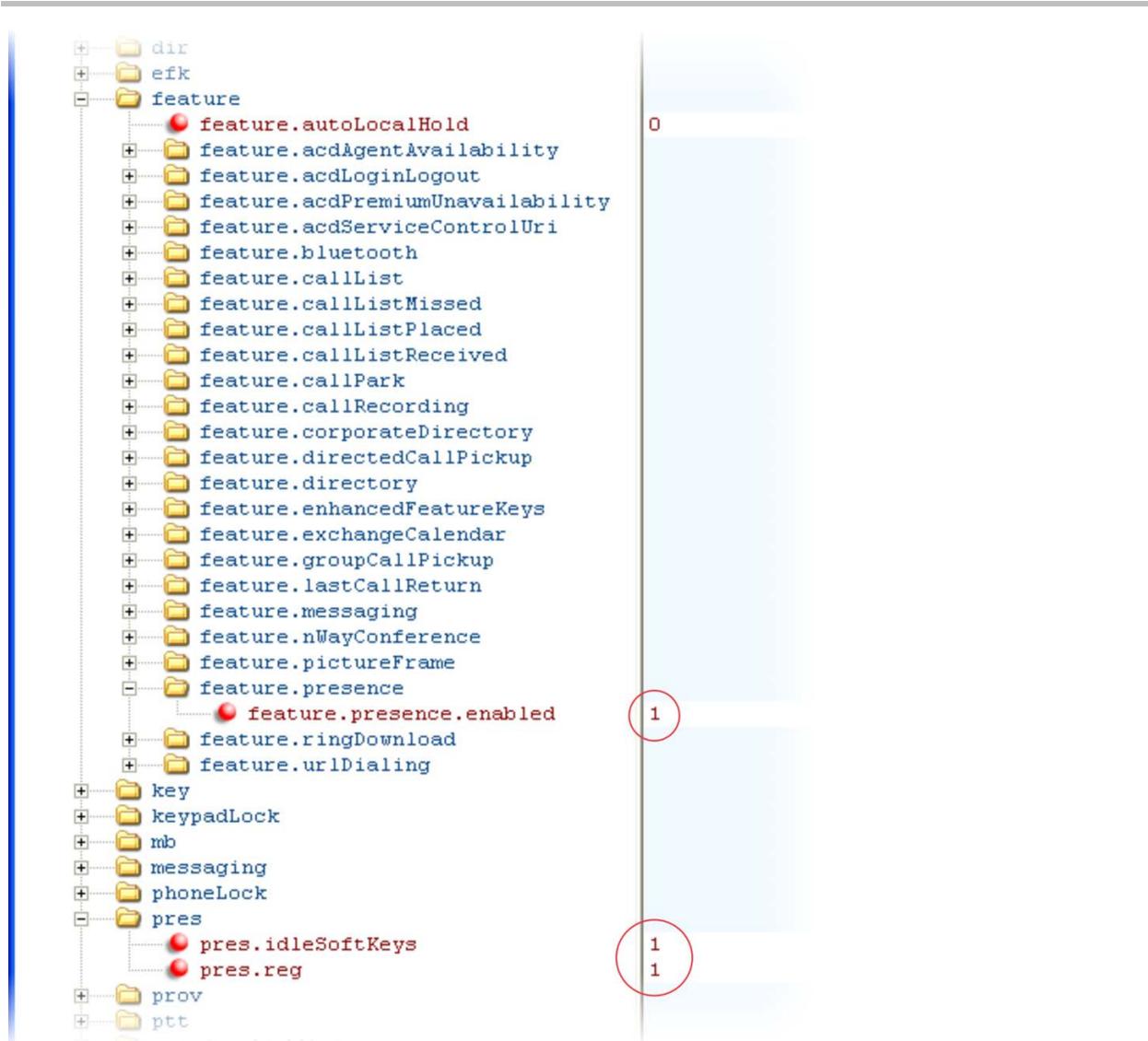
The behavior of the presence feature varies when you use Microsoft® Live Communications Server 2005 or Microsoft Office Communications Server 2007 R2 as the call server.

Table 7-33: Using the Presence Feature

| Central Provisioning Server | template > parameter |
|--|---|
| Specify the line/registration number used to send SUBSCRIBE for presence.... | features.cfg > pres.reg |
| Specify if the MyStatus and Buddies soft keys display on the Home screen | features.cfg
> pres.idleSoftkeys |
| Turn the presence feature on or off | features.cfg > feature.presence.enabled |
| Local Phone User Interface | |
| The user can edit the directory contents. The <i>Buddy Watch</i> and <i>Buddy Block</i> fields control the buddy behavior of contacts. | |

Example Presence Configuration

In the following illustration, the presence feature has been enabled in `feature.presence.enabled`. The **MyStatus** and **Buddies** soft keys will both display on the phone's home screen when you enable the `pres.idleSoftkeys` parameter. The `pres.reg` parameter will use the address of phone line 1 for the presence feature.



This configuration will enable the presence feature and display the **MyStatus** and **Buddies** soft keys on the phone. When you press the **Buddies** soft key, contacts you have entered to your Buddy List will display. In the next illustration, Lisa Wong has been entered to the Buddies List.



Using CMA Presence

The CMA Presence feature, available only on the VVX 1500 phones, enables you to monitor the status of other remote users and phones. By adding remote users to your Buddy List, you can monitor changes in the status of remote users in real time or you can monitor remote users as speed-dial contacts. Phone users can block others from monitoring their phones and receive notifications when the status of a remote line changes. Phone users can also manually specify their status in order to override or mask automatic status updates to others.

If you want to configure CMA presence for a VVX 1500 phone, you will need to provision the phone using the Polycom Converged Management Application (CMA) system. See [Provisioning VVX Phones Using a Polycom CMA System](#) in Chapter 4 of this guide.



Tip: Using the CMA Presence Feature

This feature is available on the VVX 1500 phone only and requires provisioning of the phone by a Polycom CMA system.



Web Info: Using the CMA Presence Feature

For more information on the CMA presence feature, see [User Guide for the Polycom VVX 1500 Business Media Phone](#).

Enabling Access URL in SIP Messages

When this feature is enabled, the server can attach a URL to incoming and active calls. The phone's browser or microbrowser can read this URL and render it as Web content that displays on the phone screen. This feature can be enabled on SoundPoint IP, SoundStation IP, and VVX 500 and 1500 phones (see [Table 7-33: Enabling Access URL in SIP Messages](#)).

This feature is flexible and can be used in the following ways.

- A Call Center
 - A URL is attached to an incoming call and displays extended information about a customer before the agent takes the call.
 - The phone can display a script of questions for an agent to ask a caller, and a different script can be provided to different agent groups.
- A Restaurant menu on a hotel phone
 - A guest dials a number for the restaurant or room service and a voice indicates that the menu is available for viewing on the phone.

There are three user interface aspects to this feature:

- **Web Content Status Indication** When valid Web content is available on the phone, an icon displays beside the call information. In the examples shown next, a lightning bolt icon indicates Web content is available for a call appearance. The phone user can press the Select key to display the Web content.

The following figure shows the SoundPoint IP 331 phone user interface.



The following figure shows the SoundPoint IP 550 phone user interface.



- **Web Content Retrieval** Phone users can choose to retrieve Web content in Active Mode (spontaneously) or in Passive Mode (by request).

- **Active Mode** There are two ways to configure spontaneous Web content retrieval: you can set the Web content retrieval parameter in the configuration file to 'active' or, if your call server supports access URL, you can specify active retrieval in the SIP heading. If parameters in the SIP signal conflict with the file configuration, parameters in the SIP signaling will take precedence. Note that incoming active Web content will interrupt Web content currently being viewed.
- **Passive Mode** There are two ways to configure Web content retrieval by request: you can set the Web content retrieval parameter in the configuration file to 'passive' or, if your call server supports access URL, you can specify passive retrieval in the SIP heading. When passive mode is enabled, an icon displays beside a call appearance indicating that Web content is available. For more information about the Web content icon, see Web Content Status Indication earlier in this section. When an icon shows that Web content is available, the phone user can press the Select key to view the content. If the Web content has expired, no icon displays and the Select key will perform no function. Note that incoming active Web content will interrupt Web content currently being viewed. Passive mode is recommended when the microbrowser is used for other applications.
- **Settings Menu** You can enable new Web content to be added to the phone's menu. Using the phone's menu, users can set the default display mode for individual URLs to active or passive.

You must use the following standards if you want to set the retrieval display mode of Web content in the SIP headers:

- A new SIP header must be used to report Web content associated with SIP phone calls (the SSAWC header follows the BNF for the standard SIP header Alert-Info):

```
Alert-Info = "Alert-Info" HCOLON alert-param *(COMMA alert-param)
alert-param = LAQUOT absoluteURI RAQUOT *( SEMI generic-param )
```

The Web content must be located with an absolute URI that begins with the scheme identifier. Currently only the HTTP scheme is supported.

The following is an example of a valid SIP header:

```
Access-URL: <http://server.polycom.com/content23456.xhtml>
```

This header may be placed in SIP requests and responses so long as the messages are part of an INVITE-initiated dialog and the phone can associate them with an existing phone call.

You may also define two optional parameters:

- An `expires` parameter is defined to indicate the lifespan of the URL itself. Or, if the URL is permanent, you can set how long the Web content will display with the call. If the parameter is absent or invalid, this will be interpreted to mean that the content or the URL itself will be persistent. A value, if it is present, will indicate the lifespan of the content in seconds (zero has special significance—see the next example). When the lifespan expires, the phone will remove both the indication of the URL and the ability of the user to retrieve it.

For example:

```
Access-URL:
<http://server.polycom.com/content23456.xhtml>; expires=60
```

If the server wishes to invalidate a previous URL, it can send a new header (through UPDATE) with expires=0. The expires parameter is ignored when determining whether to spontaneously retrieve the Web content unless expires=0.

- A mode parameter is defined to indicate whether the Web content should be displayed spontaneously or retrieved on-demand. Two values are allowed: active and passive. If the parameter is absent or invalid, this will be interpreted the same as passive, meaning that the Web content will be retrievable on-demand but will not be spontaneously displayed. If the value is set to active, the Web content will be spontaneously displayed, subject to the rules discussed under **Active Mode** in Web Content Retrieval earlier in this section.

For example:

```
Access-URL:
<http://server.polycom.com/content23456.xhtml>;expires=60;
mode=passive
```

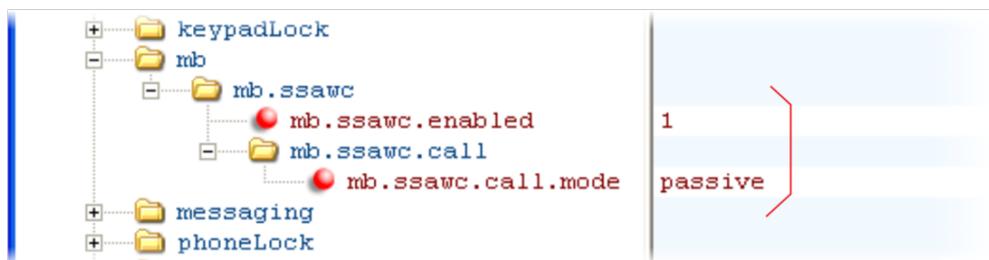
In this case, an icon will indicate that Web content is available for a period of 60 seconds.

Table 7-34: Enabling Access URL in SIP Messages

| Central Provisioning Server | template > parameter |
|-------------------------------------|-----------------------------------|
| To turn this feature on or off..... | features.cfg > mb.ssawc.enabled |
| To retrieve content | features.cfg > mb.ssawc.call.mode |

Example Access URL in SIP Messages Configuration

In the following example, in the features.cfg template, the access URL in SIP message feature is enabled in mb.ssawc.enabled . The parameter mb.ssawc.call.mode is set to passive, which means Web content will not display spontaneously; Web content will display when activated by the phone user.



Configuring the Static DNS Cache

Beginning SIP 2.1.0, failover redundancy can only be used 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.

The solution in SIP 3.1 is to enable you to statically configure a set of DNS NAPTR SRV and/or A records into the phone (see [Table 7-34: Configuring the Static DNS Cache](#)).

When a phone is configured with a DNS server, it will behave as follows by default:

- The phone will make an initial attempt to resolve a hostname that is within the static DNS cache. For example, a query will be made to the DNS if the phone registers with its SIP registrar.
- 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.
- After the configured time interval has elapsed, a resolution attempt of the hostname will again result in a query to the DNS.
- 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.

When a phone is not configured with a DNS server, it will behave as follows:

- When the phone attempts to resolve a hostname within the static DNS cache, it will always return the results from the static cache.

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 [RFC 2308](#).

Table 7-35: Configuring the Static DNS Cache

| Central Provisioning Server | template > parameter |
|---|--|
| Specify the line registration | sip_interop.cfg > reg.x.address |
| Specify the call server used for this registration | sip_interop.cfg > reg.x.server.y.* |
| Specify the DNS A address, hostname, and cache time interval (ttl)
..... | site.cfg > dns.cache.A.x.* |
| Specify the DNS NAPTR parameters, including: name, order, preference, regexp, replacement,
service, and ttl
..... | site.cfg > dns.cache.NAPTR.x.* |
| Specify DNS SRV parameters, including: name, port, priority, target, ttl, and weight
..... | site.cfg > dns.cache.SRV.x.* |

Example Static DNS Cache Configuration

The following examples show you how to configure the static DNS cache.

Example 1

This example shows how to configure static DNS cache using A records IP addresses in SIP server address fields.

When the static DNS cache is not used, the **site.cfg** configuration will look as follows:

| reg | |
|--------------------------|--------------|
| reg.1.address | 1001 |
| reg.1.server.1.address | 172.23.0.140 |
| reg.1.server.1.port | 5075 |
| reg.1.server.1.transport | UDPOnly |
| reg.1.server.2.address | 172.23.0.150 |
| reg.1.server.2.port | 5075 |
| reg.1.server.2.transport | UDPOnly |

When the static DNS cache is used, the **site.cfg** configuration will look as follows:

| reg | |
|--------------------------|-----------------------|
| reg.1.address | 1001 |
| reg.1.server.1.address | sipserver.example.com |
| reg.1.server.1.port | 5075 |
| reg.1.server.1.transport | UDPOnly |
| reg.1.server.2.address | |
| reg.1.server.2.port | |
| reg.1.server.2.transport | |
| dns.cache.A.1.name | sipserver.example.com |
| dns.cache.A.1.ttl | 3600 |
| dns.cache.A.1.address | 172.23.0.140 |
| dns.cache.A.2.name | sipserver.example.com |
| dns.cache.A.2.ttl | 3600 |
| dns.cache.A.2.address | 172.23.0.150 |



Note: Details of the Preceding Example

Above addresses are presented to Polycom UC Software in order, for example, dns.cache.A.1, dns.cache.A.2, and so on.

Example 2

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 will look as follows:

| Setting | Value |
|---------------------------------------|--|
| <code>reg.1.address</code> | <code>1002@sipserver.example.com</code> |
| <code>reg.1.server.1.address</code> | <code>primary.sipserver.example.com</code> |
| <code>reg.1.server.1.port</code> | <code>5075</code> |
| <code>reg.1.server.1.transport</code> | <code>UDPOnly</code> |
| <code>reg.1.server.2.address</code> | <code>secondary.sipserver.example.com</code> |
| <code>reg.1.server.2.port</code> | <code>5075</code> |
| <code>reg.1.server.2.transport</code> | <code>UDPOnly</code> |

When the static DNS cache is used, the **site.cfg** configuration will look as follows:

| Setting | Value |
|---------------------------------------|--|
| <code>reg.1.address</code> | <code>1002</code> |
| <code>reg.1.server.1.address</code> | <code>sipserver.example.com</code> |
| <code>reg.1.server.1.port</code> | |
| <code>reg.1.server.1.transport</code> | <code>UDPOnly</code> |
| <code>reg.1.server.2.address</code> | |
| <code>reg.1.server.2.port</code> | |
| <code>reg.1.server.2.transport</code> | |
| <code>dns.cache.SRV.1.name</code> | <code>_sip._udp.sipserver.example.com</code> |
| <code>dns.cache.SRV.1.ttl</code> | <code>3600</code> |
| <code>dns.cache.SRV.1.priority</code> | <code>1</code> |
| <code>dns.cache.SRV.1.weight</code> | <code>1</code> |
| <code>dns.cache.SRV.1.port</code> | <code>5075</code> |
| <code>dns.cache.SRV.1.target</code> | <code>primary.sipserver.example.com</code> |
| <code>dns.cache.SRV.2.name</code> | <code>_sip._udp.sipserver.example.com</code> |
| <code>dns.cache.SRV.2.ttl</code> | <code>3600</code> |
| <code>dns.cache.SRV.2.priority</code> | <code>2</code> |
| <code>dns.cache.SRV.2.weight</code> | <code>1</code> |
| <code>dns.cache.SRV.2.port</code> | <code>5075</code> |
| <code>dns.cache.SRV.2.target</code> | <code>secondary.sipserver.example.com</code> |



Settings: Port Value Settings

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 3

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 used, the **site.cfg** configuration will look as follows:

| | |
|--------------------------|----------------------------|
| reg | |
| reg.1.address | 1002@sipserver.example.com |
| reg.1.server.1.address | 172.23.0.140 |
| reg.1.server.1.port | 5075 |
| reg.1.server.1.transport | UDPOnly |
| reg.1.server.2.address | 172.23.0.150 |
| reg.1.server.2.port | 5075 |
| reg.1.server.2.transport | UDPOnly |

| | |
|--------------------------|----------------------------|
| reg | |
| reg.1.address | 1002@sipserver.example.com |
| reg.1.server.1.address | 172.23.0.140 |
| reg.1.server.1.port | 5075 |
| reg.1.server.1.transport | UDPOnly |
| reg.1.server.2.address | 172.23.0.150 |
| reg.1.server.2.port | 5075 |
| reg.1.server.2.transport | UDPOnly |

When the static DNS cache is used, the `site.cfg` configuration will look as follows:

| | |
|-------------------------------|---------------------------------|
| site | |
| reg.1.address | 1002 |
| reg.1.server.1.address | sipserver.example.com |
| reg.1.server.1.port | |
| reg.1.server.1.transport | |
| reg.1.server.2.address | |
| reg.1.server.2.port | |
| reg.1.server.2.transport | |
| dns.cache.NAPTR.1.name | sipserver.example.com |
| dns.cache.NAPTR.1.ttl | 3600 |
| dns.cache.NAPTR.1.order | 1 |
| dns.cache.NAPTR.1.preference | 1 |
| dns.cache.NAPTR.1.flag | s |
| dns.cache.NAPTR.1.service | SIP+D2U |
| dns.cache.NAPTR.1.regexp | |
| dns.cache.NAPTR.1.replacement | _sip._udp.sipserver.example.com |
| dns.cache.SRV.1.name | _sip._udp.sipserver.example.com |
| dns.cache.SRV.1.ttl | 3600 |
| dns.cache.SRV.1.priority | 1 |
| dns.cache.SRV.1.weight | 1 |
| dns.cache.SRV.1.port | 5075 |
| dns.cache.SRV.1.target | primary.sipserver.example.com |
| dns.cache.SRV.2.name | _sip._udp.sipserver.example.com |
| dns.cache.SRV.2.ttl | 3600 |
| dns.cache.SRV.2.priority | 2 |
| dns.cache.SRV.2.weight | 1 |
| dns.cache.SRV.2.port | 5075 |
| dns.cache.SRV.2.target | secondary.sipserver.example.com |
| dns.cache.A.1.name | primary.sipserver.example.com |
| dns.cache.A.1.ttl | 3600 |
| dns.cache.A.1.address | 172.23.0.140 |
| dns.cache.A.2.name | secondary.sipserver.example.com |
| dns.cache.A.2.ttl | 3600 |
| dns.cache.A.2.address | 172.23.0.150 |



Settings: Forcing NAPTR Lookups

The `reg.1.server.1.port`, `reg.1.server.2.port`, `reg.1.server.1.transport`, and `reg.1.server.2.transport` values in this example are set to null to force NAPTR lookups.



Web Info: Using a Static DNS Cache

For more information about using a static DNS cache, see [Technical Bulletin 36033: Using a Static DNS Cache with SoundPoint IP and SoundStation IP Phones](#).

Displaying SIP Header Warnings

The warning field from a SIP header may be configured to display a three second pop-up message on the phone, for example, that a call transfer failed due to an invalid extension number. For more information, see [Header Support](#).

You can display these pop-up messages in any language supported by the phone. The messages will display for three seconds unless overridden by another message or action. To turn the warning display on or off or specify which warnings are displayable, you can configure the parameters in [Table 7-35: Displaying SIP Header Warnings](#).

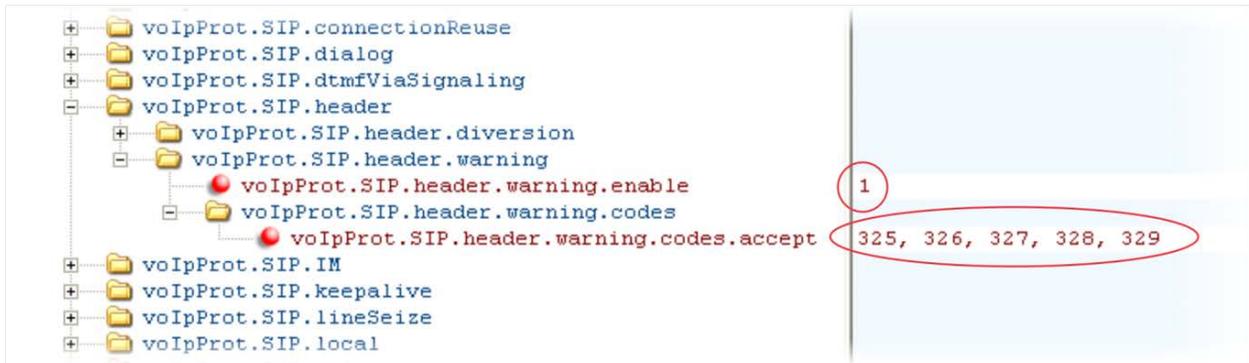
Table 7-36: Displaying SIP Header Warnings

| Central Provisioning Server | template > parameter |
|---|--|
| Turn this feature on or off..... | <code>sip-interop.cfg > volpProt.SIP.header.warning.enable</code> |
| Specify which warnings are displayable .. | <code>sip-interop.cfg > volpProt.SIP.header.warning.codes.accept</code> |

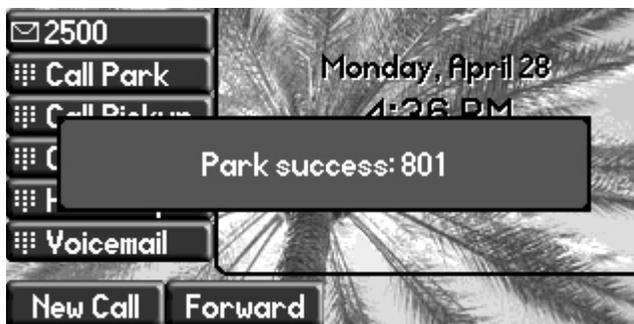
Example Display of Warnings from SIP Headers Configuration

To enable the display of warnings from SIP headers, set the `voIpProt.SIP.header.warning.enable` parameter in the `features.cfg` template to 1. Enter the warning codes as a comma-separated string. The strings associated with the values 325 to 329 that display on the phone screen, as shown in the next illustration, have been entered automatically by the call server and are not entered by the administrator in the configuration file.

The following illustration shows a sample configuration from the `sip-interop.cfg` template file:



The next illustration shows a SIP header message displayed on a SoundPoint IP 550 phone.



Quick Setup of Polycom Phones

A Quick Setup feature was added to simplify the process of entering the provisioning (boot) server parameters from the phone's user interface. This feature is designed to make it easier for on-site *out of the box* provisioning of SoundPoint IP, SoundStation IP, and VVX phones and SpectraLink handsets.

When you enable this feature, a **QSetup** soft key will display on the phone (see [Table 7-36: Quick Setup of Polycom Phones](#)). When you press the **QSetup** soft key, a new menu will display. The menu enables you to access the provisioning server and quickly configure the phone to work. After configuring the Quick Setup, you can disable display of the **QSetup** soft key using a configuration file setting.

You can enable the Quick Setup feature through the **site.cfg** configuration file or through the phone's menu.



Web Info: [Configuring Quick Setup](#)

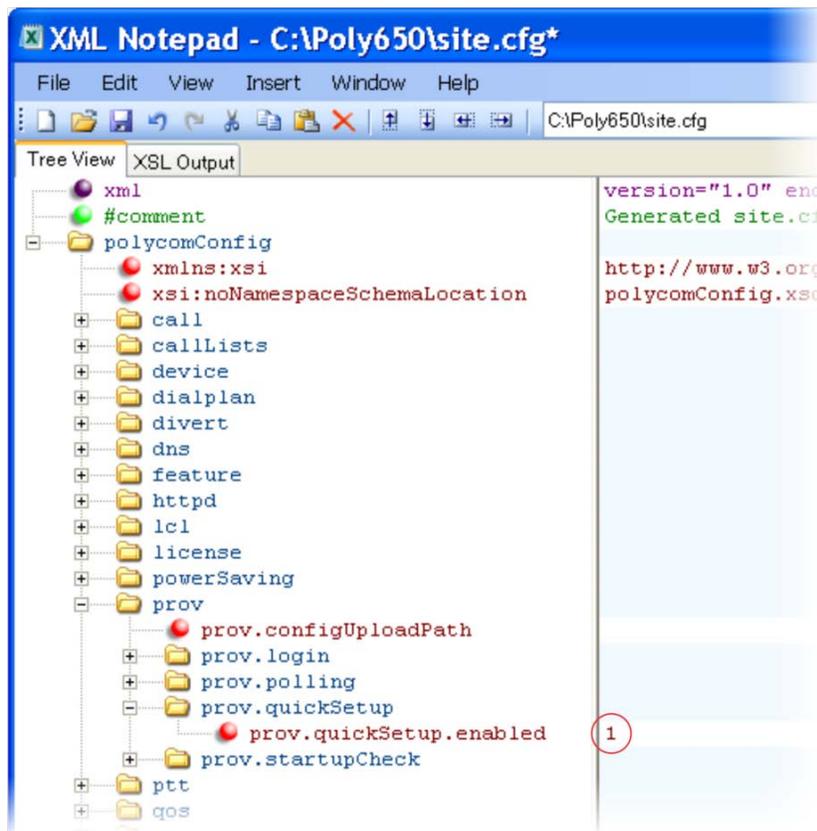
For details on how to configure quick setup, see [Technical Bulletin 45460: Using Quick Setup with Polycom Phones](#).

Table 7-37: Quick Setup of Polycom Phones

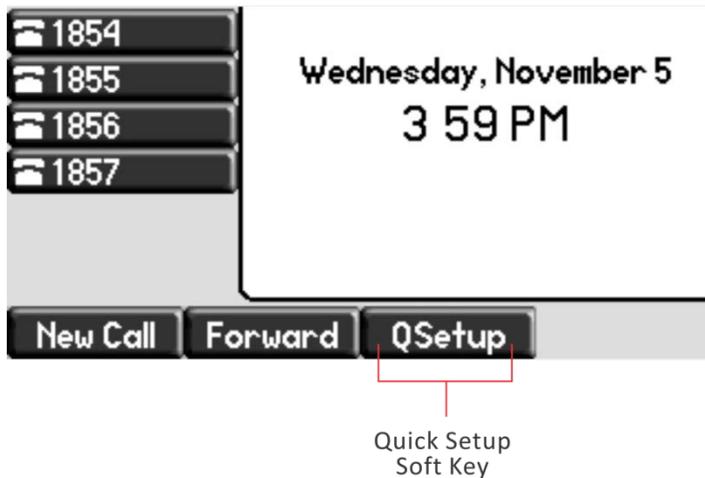
| | |
|---------------------------------------|--|
| Central Provisioning Server | template > parameter |
| To enable or disable Quick Setup..... | site.cfg > prov.quickSetup.enabled |

Example Quick Setup Configuration

To enable the Quick Setup feature, enable the `prov.quickSetup.enabled` parameter in the `site.cfg` template file, shown next.



The **QSetup** will display on the phone screen, shown next.



Press the **QSetup** soft key to open the menu and access the quick setup feature.

Provisional Polling of Polycom Phones

You can configure how your phone provisioning automatically by configuring the parameters in [Table 7-37: Provisional Polling of Polycom Phones](#).

You can set the phone's automatic provisioning behavior to be:

- **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 time interval you set.
 - If the time period is less than a day or equal to one day, the first poll is at a random time between the phone starting up and the polling period. Afterwards, the phone will poll every x seconds.
 - If you set the polling period to be greater than one day, the phone polls on a random day based on the phone's MAC address.

For example:

- 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 `86400`, `prov.polling.time` is set to `01:00`, `prov.polling.timeRandomEnd` is set to `05:00`, the phone polls randomly between 1am and 5am every day.

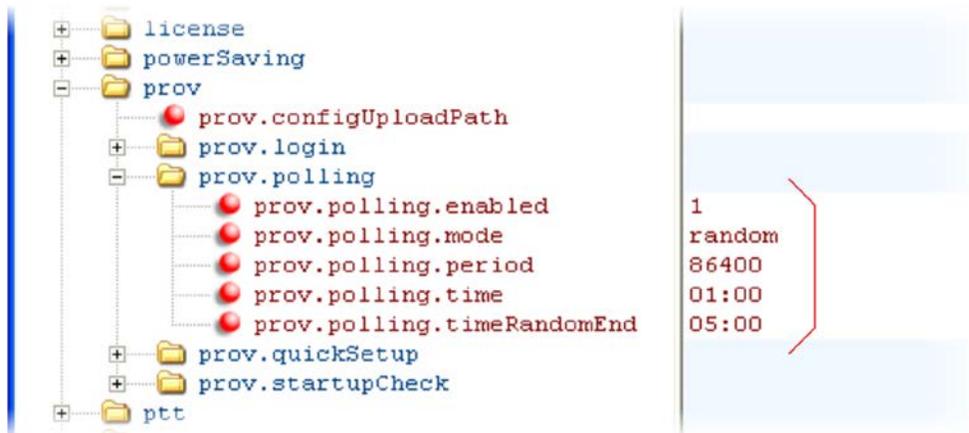
- If `prov.polling.mode` is set to `abs` and `prov.polling.period` is set to `2328000`, the phone polls every 20 days.

Table 7-38: Provisional Polling of Polycom Phones

Central Provisioning Server
template > parameterTo enable polling and set the mode, period, time, and time end parameters **site.cfg > prov.polling.***

Example Provisional Polling Configuration

The following illustration shows the default sample random mode configuration for the provisional polling feature in the **site.cfg** template file.



Setting Up Microsoft Office Communications Server 2007 R2 Integration

This feature is supported only on the SpectraLink 8400 Series handsets. You can use Microsoft Office Communications Server (OCS) 2007 R2 to help improve efficiency and increase productivity and to share ideas and information immediately with business contacts. Use the parameters in [Table 7-39: Setting Up Microsoft Office Communications Server 2007 R2 Integration](#) to configure this feature.

Table 7-39: Setting Up Microsoft Office Communications Server 2007 R2 Integration

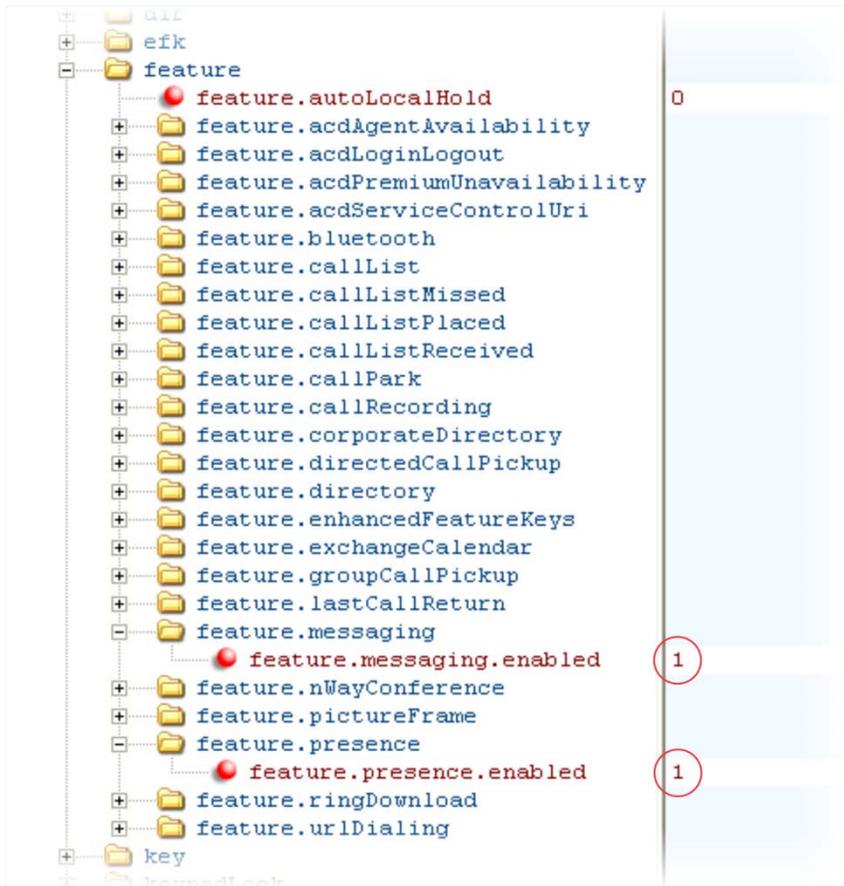
| Central Provisioning Server | template > parameter |
|---|---|
| Enable support for Microsoft Office Communications Server 2007 R2 | sip-interop.cfg > volpProt.SIP.lcs |
| Enable or disable SIP forking..... | sip-interop.cfg > volpProt.SIP.ms-forking |
| Specify the line/registration number used to send SUBSCRIBE for presence.... | features.cfg > pres.reg |
| Turn the presence feature on or off | features.cfg > feature.presence.enabled |
| Turn the messaging feature on or off..... | features.cfg > feature.messaging.enabled |
| Specify the line/registration number which has roaming buddies support enabled
..... | features.cfg > roaming_buddies.reg |
| Specify the line/registration number which has roaming privacy support enabled
..... | features.cfg > roaming_privacy.reg |
| Specify the number of line keys to use for a single registration | reg-advanced.cfg > reg.x.lineKeys |

Example OCS 2007 R2 Integration Configuration (Single Registration)

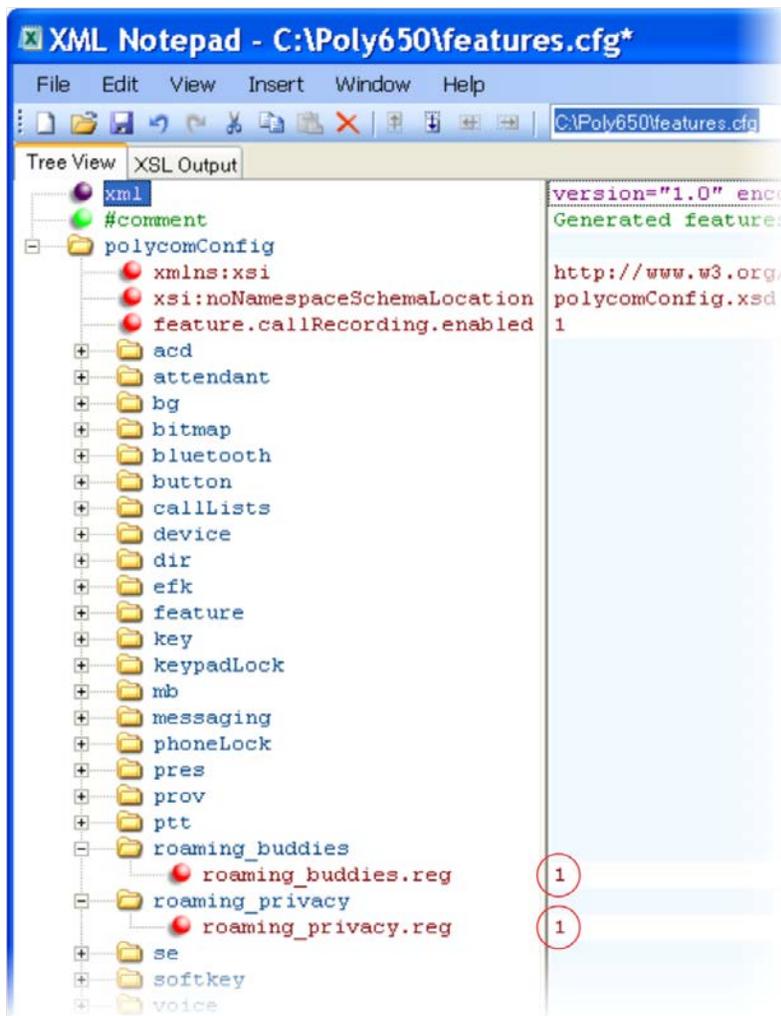
The following examples show you the minimum configuration you need to set to integrate a phone that has a single registration with OCS 2007 R2.

In **features.cfg**, you will need to enable the presence and messaging features, as well as set a line/registration number on which roaming buddies and roaming privacy support is enabled.

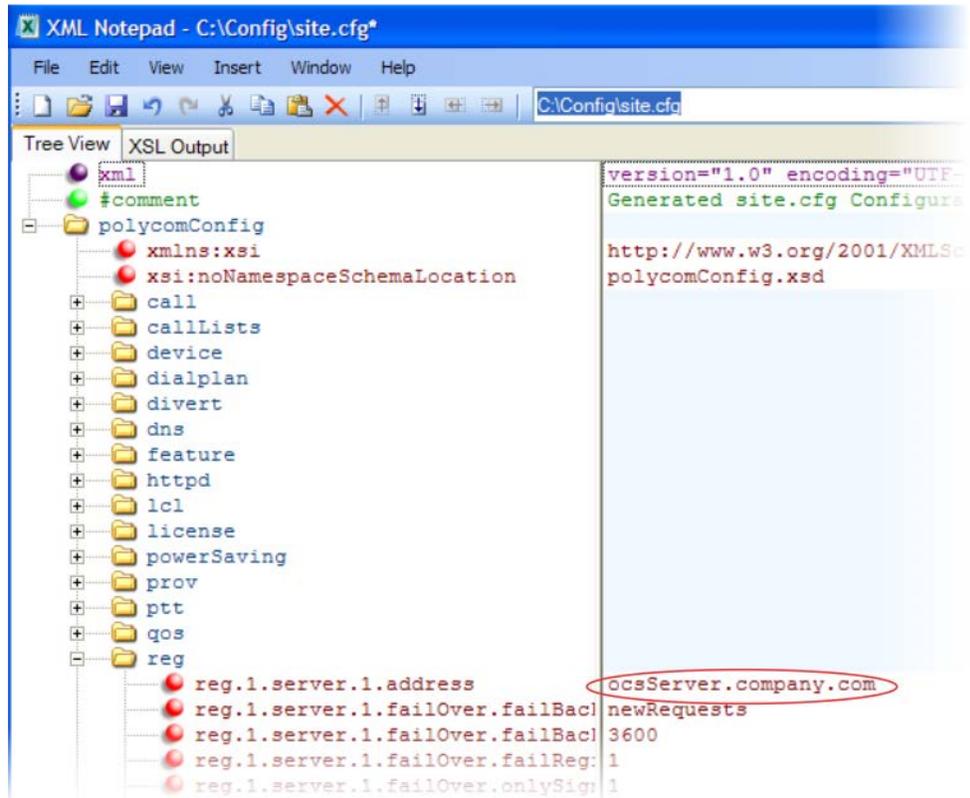
The following example shows how to enable the presence and messaging features.



The following example shows you how to enable the roaming buddies and roaming privacy parameters.

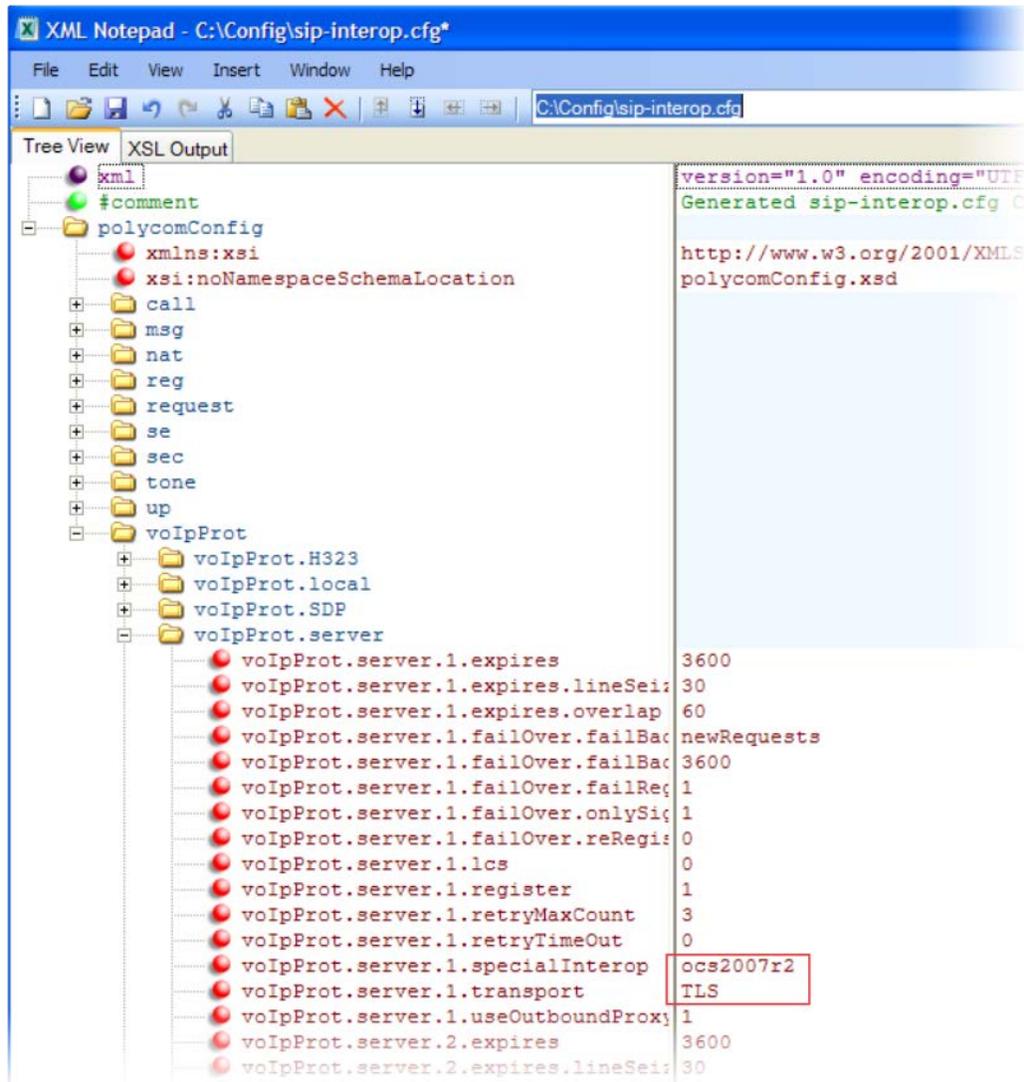


Next, specify the OCS 2007 R2 server name in **site.cfg**, as shown next.

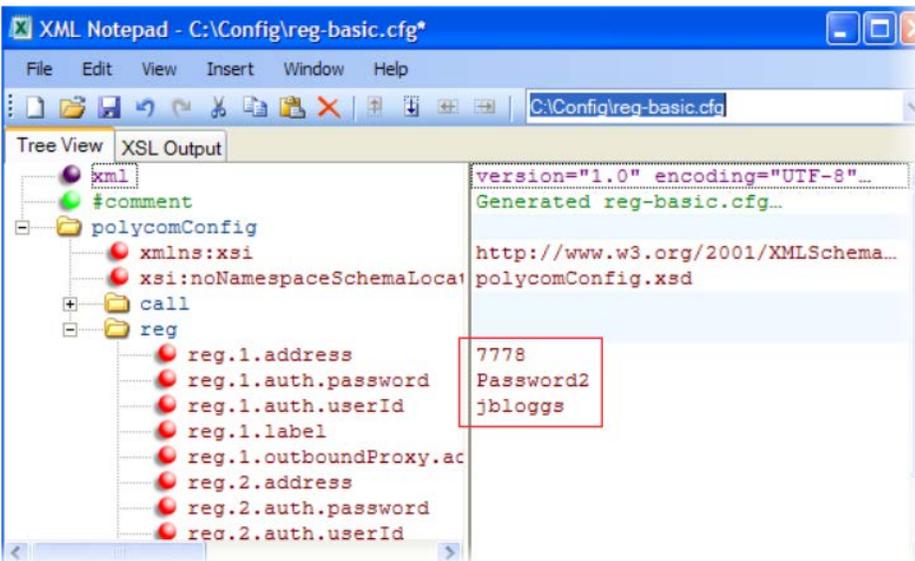


Then, in the **sip-interop.cfg** template:

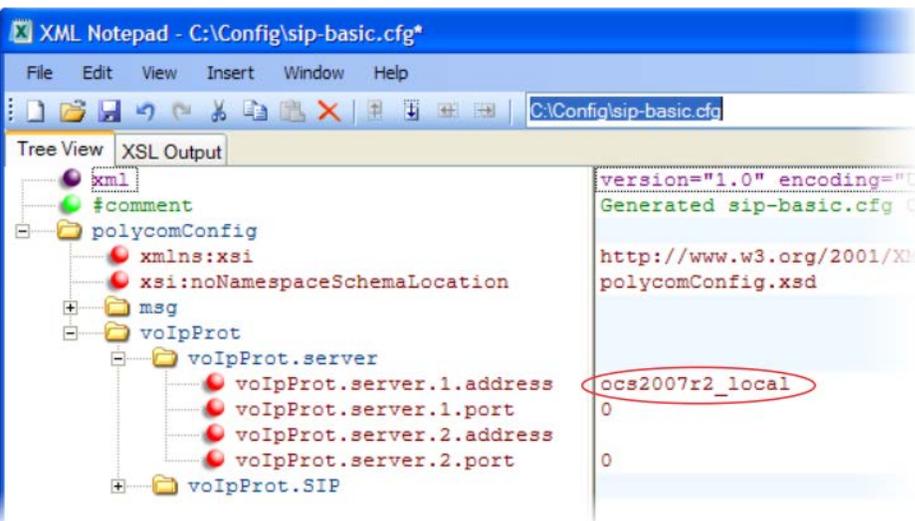
- For the server type parameter (`voIpProt.server.1.specialInterop`), specify `ocs2007r2`. You must assign this value for instant messaging to work.
- Specify the `transport` protocol for OCS 2007 R2 to use. For the transport protocol parameter (`voIpProt.server.1.transport`), specify either `TCPPreferred` or `TLS`, depending on your phone environment.



Next, in **reg-basic.cfg**, specify the OCS 2007 R2 address, and a phone's OCS 2007 R2 user ID and password, as shown next.



Finally, configure the OCS 2007 R2 address in **sip-basic.cfg**, as shown next.



Setting Up Microsoft Lync Server 2010 Integration

Microsoft® Lync® Server 2010 is a unified communications (UC) solution that enables customers, colleagues, and business partners to communicate instantly by voice, video, or messaging through a single interface, regardless of their location or network. You can register with Lync Server 2010 and use Lync features on Polycom SoundPoint IP 321, 331, 335, 450, 550, 560, and 650 phones, VVX 500 and 1500 business media phones, SoundStation IP 5000,

SoundStation Duo, and SpectraLink 8400 series handsets. Note that you must use Polycom UC Software 4.1.0 to register phones with Lync Server 2010. For detailed instructions on updating the phone to Polycom UC Software 4.1.0, see the [Polycom Lync Provisioning Guide](#). Note also that you must purchase a Lync Feature License from a Polycom reseller or Polycom sales representative.



Note: You must purchase a license to use Microsoft Lync Server 2010 with Polycom phones.

You must purchase a *Lync Feature License* from a Polycom reseller or Polycom sales representative to use Polycom SoundPoint IP, SoundStation IP, and VVX products in a Microsoft Lync environment. This license is not required for SpectraLink 8400 Series wireless handsets. You can use Polycom phones in a Lync environment for trial purposes, without purchasing a license, to a maximum of 30 days.

For details on the features available on Polycom phones provisioned with Microsoft Lync Server 2010, see [Feature Profile 72430: Using Polycom Phones with Microsoft Lync Server 2010](#). The concurrent failover/fallback feature, explained in [Setting Up Server Redundancy](#), is not compatible with Microsoft Lync. Polycom UC Software enables you to register a single phone line with Lync Server; you cannot register shared lines with Lync Server.



Note: Understanding the Lync Contact List and Your Phone's Local Contact Directory

When you are running UC Software 4.1.x for use with Lync Server 2010, you have access to two separate contact lists: the default local contact directory on your Polycom phone and a Lync contact list. If you want to disable the local contact directory on your Polycom phone or make it read-only, see [Using the Local Contact Directory](#).

Registering with Microsoft Lync Server 2010

You can register Polycom phones with Lync Server 2010 using one of three ways:

- Using the Web Configuration Utility
- Using centralized provisioning, which includes a provisioning server and configuration files in XML format.
- From the phone user interface



Note: Registering a Phone with Lync Server 2010

For details on using the phone user interface and for details on each registration method, including registration instructions, see the [Polycom Lync Provisioning Guide](#).

Setting the Base Profile to Lync - Phone User Interface and Web Configuration Utility

Once you have updated your phones to use Polycom UC Software 4.1.0, you can quickly register phones with the Lync Server by setting the phone's Base Profile to *Lync* from the phone's user interface or using the Web Configuration Utility. Note that although registering the phone using either of these two methods is simpler than centralized provisioning, each method registers one phone at a time. In addition, you cannot enable extensive diagnostic logging that the phone writes to the provisioning server, contact directory files, or phone user interface language files. You can use the Web Configuration Utility to update a phone to UC Software 4.1.0.

Centralized Provisioning

You can update to Polycom UC Software 4.1.0 and register multiple Polycom phones to Lync Server using a provisioning server and configuration files in XML format. You can provision your phones with Lync Server 2010 using the **lync.cfg** template configuration file included with Polycom UC Software 4.1.0. Polycom recommends using this method - also called centralized provisioning - when deploying multiple phones, about twenty or more. A provisioning server enables you to store configuration files in a single location on a server, which simplifies maintenance of feature settings and updates for multiple phones. In addition, use of a provisioning server allows the phones to send diagnostic and other information to files stored on the server, including log files, a contact directory, individual call lists, and multiple languages on the phone user interface. Note that you must use a provisioning server to update your phones to Polycom UC Software 4.1.0, which is required to register phones with Lync Server.

Example Configuration: Setting the Base Profile to Lync

This example configuration shows you how to set the phone's Base Profile to *Lync* from the phone idle screen using the multi-key combo (MKC) shortcut. For instructions on all methods you can use to provision Polycom phones with Lync Server, including tips on how to quickly provision multiple phones to save time, see the [Polycom Lync Provisioning Guide](#).

When you set the phone Base Profile to Lync you are provisioning the phone with the minimum number of parameters required to register a Polycom phone with Lync Sever 2010. All of the required parameters are included in the **lync.cfg** template configuration file included with Polycom UC Software 4.1.0. This template configuration file contains the values that register the phone with Lync Server - you do not need to change the values of the Lync Base Profile parameters. However, if your organization's security procedures don't allow you to enter user IDs and password in cleartext to configuration files, you will need to set `reg.x.auth.useLoginCredentials` to 1 and instruct each user to enter their credentials through the phone's user interface—the Login Credential screen.

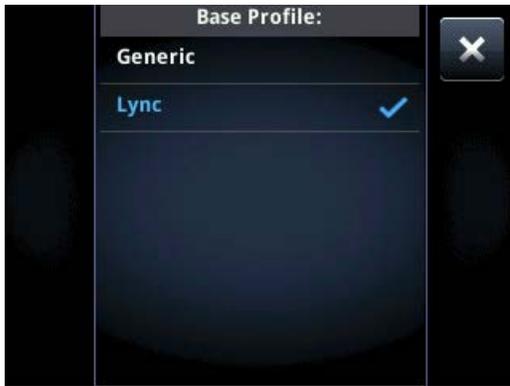
To set the Base Profile to Lync using the multi-key combo shortcut:

- 1 Press the phone's **Home/Menu** key.
- 2 From the idle screen, press and hold the following key combination on the phone keypad for about 3 seconds. These multi-key combo (MKC) keys vary by phone.

- Polycom SoundPoint IP 550, 560, and 650 desktop phones: **5, 7, 8, ***
- VVX 500 and 1500 business media phones; SpectraLink 8400 Series wireless handset: **1, 4, 9**
- SoundPoint IP 321, 331, 335, 450 desktop phones; SoundStation 5000, SoundStation Duo conference phones: **1, 2, 4, 5**

Pressing and holding the MKC keys causes the Base Profile *Password* menu to display.

- 3** In the Base Profile Password screen, enter the password (default 456) and press **Enter**.
- 4** In the *Base Profile* menu, select **Lync**.



The phone automatically restarts and displays the Lync Server *Sign In* screen.

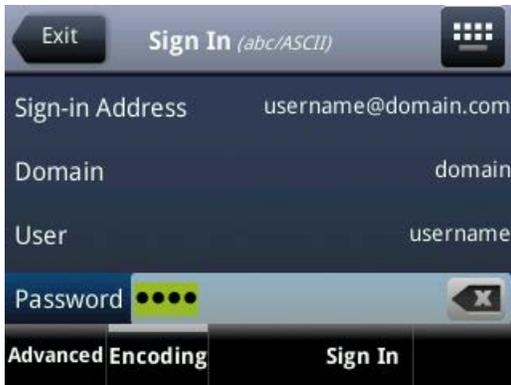


Troubleshooting: Rebooting the Phone

If the phone does not restart, you can manually restart by powering off/on the phone. You can also manually reboot the phone: Press the **Menu/Home key > Settings > Advanced**, enter the password (default 456), press **Enter**, and choose **Reboot Phone**. When the phone completes the reboot cycle, the Lync Server *Sign In* screen displays.

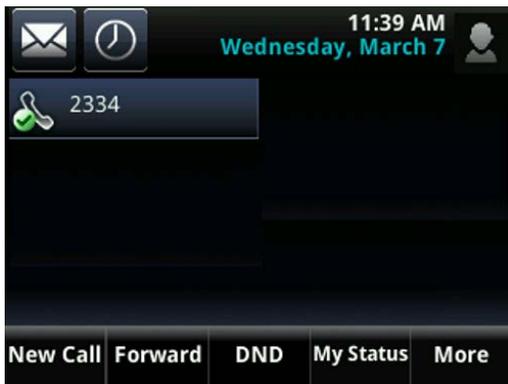
To sign in and register a line with Lync Server:

- 5** Enter your sign in credentials in the following formats:
 - **Sign In Address** This is your Lync SIP URI address, not the user name for the Active Directory account. For example, *username@domain.com*.
 - **Domain** By default, use the NetBIOS *domain* name. If that does not work, try the DNS domain name (for example, *domain.com*).
 - **User** *user name*
 - **Password** *password*



6 Select **Sign In**.

The phone registers with Lync Server and you can begin using Lync features directly from the phone. The following illustration shows a line 1, extension 2334 on the VVX 500 successfully registered to Lync Server.



7 There are two ways to sign in/out of Lync:

- a Press **Home/Menu** and go to **Settings > Features > Microsoft Lync > Sign In/Sign Out**.
- b Press the **More** soft key and select the **Sign In/Sign Out** soft key.



Admin Tip: Workaround for Phones using G.722 and Retrieving Microsoft Lync Voicemail

If your Polycom phones are configured with G.722 and users find that they do not hear audio when retrieving voicemail from the Microsoft Lync Server, you will need to make the following changes to parameters in the **site.cfg** template file:

- Change `voice.codecPref.G7221.24kbps` from **0** to **5**.
- Change `voice.codecPref.G7221.32kbps` from **5** to **0**.
- Add `voice.audioProfile.G7221.24kbps.payloadType` and set it to **112**.

Enabling Polycom Desktop Connector Integration

With the Polycom® Desktop Connector™ application installed on a computer, you can use your mouse and keyboard to enter information and navigate screens on your VVX phone running Polycom UC Software 4.0.1. This feature enables users to enter phone numbers or to select screen objects without having to use the phone's keypad or touch screen. To use this feature, the phone and computer must be on the same network or directly connected through the phone's PC port.

You will need to download and install the [Polycom Desktop Connector application](#). The Polycom Desktop Connector is compatible with computers running Microsoft® Windows XP®, Windows Vista®, and Windows® 7.

Once Polycom Desktop Connector is installed, you will need to *pair* the VVX phone and the computer (to configure the connection). If they are directly connected, there is no need to enter the VVX phone's IP address; just press the **Reconnect** soft key. If they are connected through a switch or hub, you will need to enter the computer's IP address using the phone's user interface, and press the **Reconnect** soft key. You can also change the configuration by manually editing the phone's configuration files or by using the Web Configuration Utility (see [Table 7-41: Enabling Polycom Desktop Connector Integration](#)).



Web Info: Installing and Enabling the Polycom Desktop Connector Application

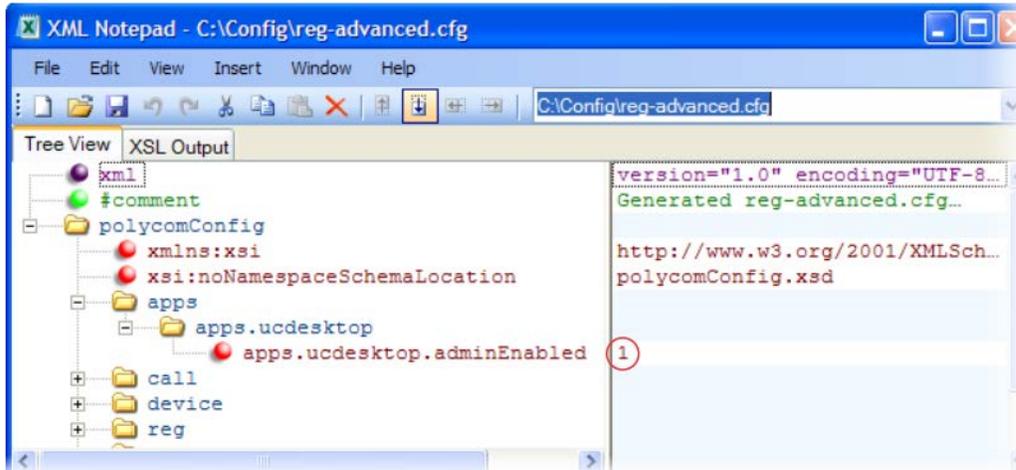
For details on how to install Polycom Desktop Connector application and enable it for use on VVX 1500 phones, see [Technical Bulletin 52855: Using Polycom Desktop Connector with Polycom VVX 1500 Phones](#).

Table 7-40: Enabling Polycom Desktop Connector Integration

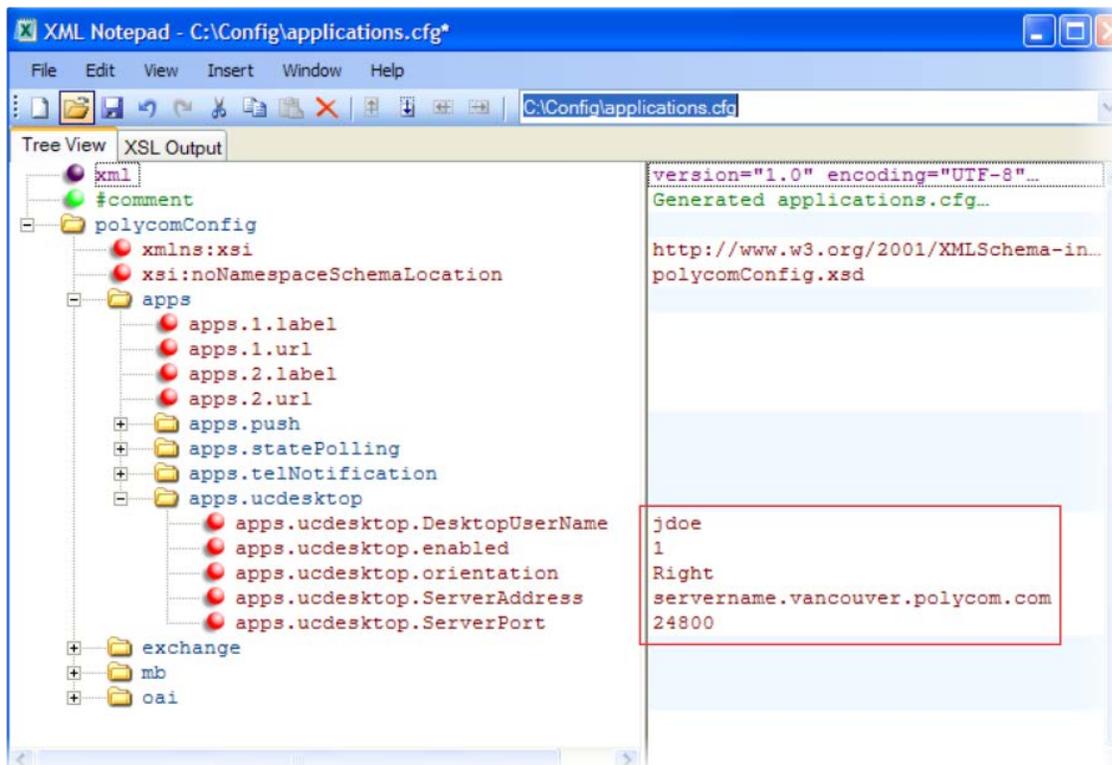
| Central Provisioning Server | template > parameter |
|---|---|
| Turn the desktop connector on or off for administrators | applications.cfg |
| > apps.ucdesktop.adminEnabled | |
| Specify the user name of the user's computer .. | applications.cfg > apps.ucdesktop.desktopUserName |
| Turn the desktop connector on or off for users | applications.cfg > apps.ucdesktop.enabled |
| Specify if the phone is positioned to the left or right of your computer | applications.cfg > apps.ucdesktop.orientation |
| Specify the server address of the user's computer.. | applications.cfg > apps.ucdesktop.ServerAddress |
| Specify the server port number for the connection | applications.cfg > apps.ucdesktop.ServerPort |
| <hr/> | |
| Web Configuration Utility | |
| To enable the user's computer to access their VVX 1500 phone, navigate to Settings > Applications and expand the Polycom Desktop Connector Client section. | |

Example PDC Configuration

To use the PDC feature, ensure that the `apps.ucdesktop.adminEnabled` in the **applications.cfg** template parameter is enabled, as shown next. By default, the parameter is enabled.



The following illustration shows the parameters in **applications.cfg** that you will need to configure to use the PDC feature on your phones. You'll have to enable the feature, as well as specify a user name, server address and port, and specify the phone's position relative to your computer.



Enabling Microsoft Exchange Calendar Integration

As of UC Software 4.0.1, VVX phones and SpectraLink handsets can display the Microsoft Exchange 2007 and 2010 calendar. The calendar gives you quick access to meeting information and you can dial in to conference calls. To integrate the Microsoft Exchange Calendar features with your phone, configure the parameters in [Table 7-42: Enabling Microsoft Exchange Calendar Integration](#)).

You can launch the feature from a calendar widget that displays in the status bar on the VVX phone. Or, you can access the feature from the **Applications** menu on the SpectraLink handsets.

You will need a valid Microsoft Windows credentials to access the Microsoft Exchange Calendar information on the phone. You can manage these credentials through the Login Credentials, which are available through **Menu > Settings > Basic > Login Credentials**.

You can view the calendar information in day or month format. On VVX phones, the meeting details displays beside the calendar view. On the SpectraLink handsets, the meeting details overlap the calendar view.

All possible phone numbers that you can dial to place a call to the meeting will display in the meeting details. You can automatically place a call by pressing a soft key.

A reminder pop-up is displayed 15 minutes before a scheduled meeting. You can dismiss the reminder, select snooze to have the reminder pop up again, open the meeting details view. A tone will be played along with the reminder pop-up.



Web Info: Using Microsoft Exchange Calendar Integration

For user instructions on how to use calendar integration, refer to the user guide for your phone:

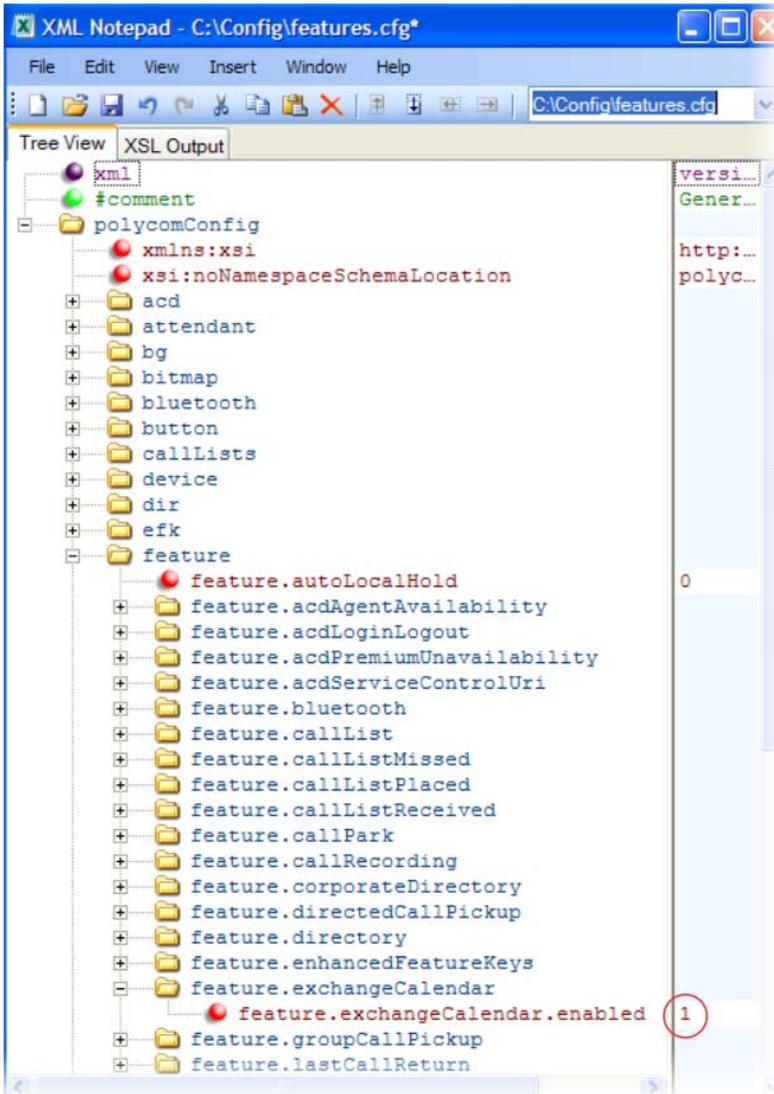
- [User Guide for the Polycom VVX 1500 Phone](#)
- [Polycom SpectraLink 8400 Series Wireless Handset User Guide](#)
- [Polycom VVX 500 Business Media Phone User Guide](#)

Table 7-41: Enabling Microsoft Exchange Calendar Integration

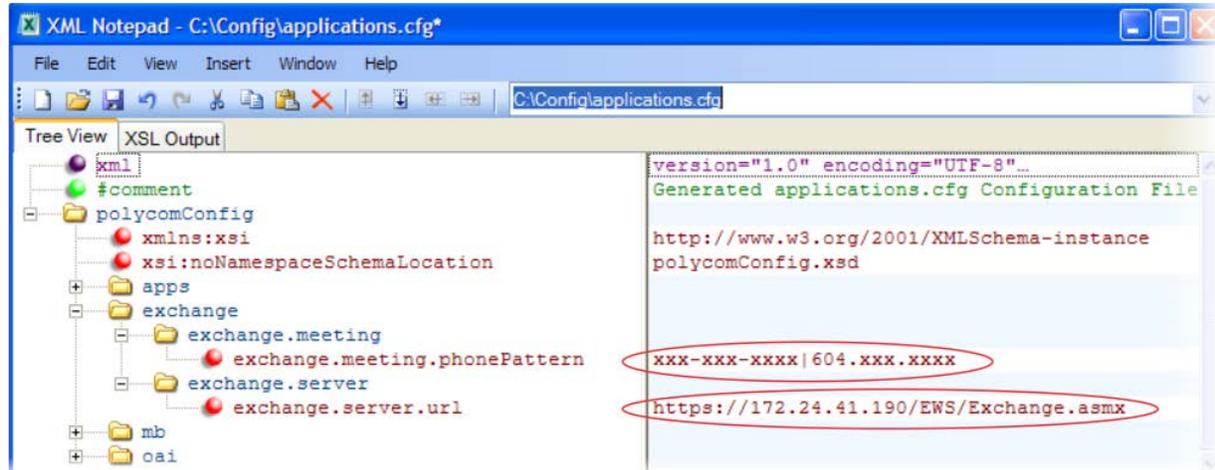
| Central Provisioning Server | template > parameter |
|---|--|
| Turn Microsoft Exchange Calendar Integration on or off | features.cfg |
| > feature.exchangeCalendar.enabled | |
| Specify the Microsoft Exchange Server address | applications.cfg > exchange.server.url |
| Specify the pattern to use to identify phone numbers in meeting descriptions | applications.cfg > exchange.meeting.phonePattern |
| Turn the meeting reminder on or off | applications.cfg > exchange.meeting.reminderEnabled |
| Web Configuration Utility | |
| To enable Microsoft Exchange Calendar Integration and configure the settings, navigate to Settings > Applications and expand Exchange Applications . | |

Example Exchange Calendar Configuration

The following example shows the Calendar feature enabled in **features.cfg**.



After you enable the feature, specify the Microsoft Exchange Server address in **applications.cfg**, as shown next. In this example, a pattern has been specified for meeting numbers. When you specify a pattern, any number in your meeting invitation that matches the pattern will display on a meeting participants' phones as a soft key. Then, participants can press the soft key to dial in to the meeting. You can specify multiple patterns, separated by a bar. In the following example, two patterns are specified.



Configuring the Polycom Quick Barcode Connector Application

If you are using SpectraLink 8450 handsets, the Polycom® Quick Barcode Connector™ (QBC) application enables you to capture and decode barcode patterns with the phone and transfer the data to applications running on one or more host computers. Data can be transferred in single endpoint mode (one host computer) or multiple endpoint mode (many host computers). To enable and configure the QBC application, configure the parameters in [Table 7-43: Configuring the Polycom Quick Barcode Connector Application](#).

Table 7-42: Configuring the Polycom Quick Barcode Connector Application

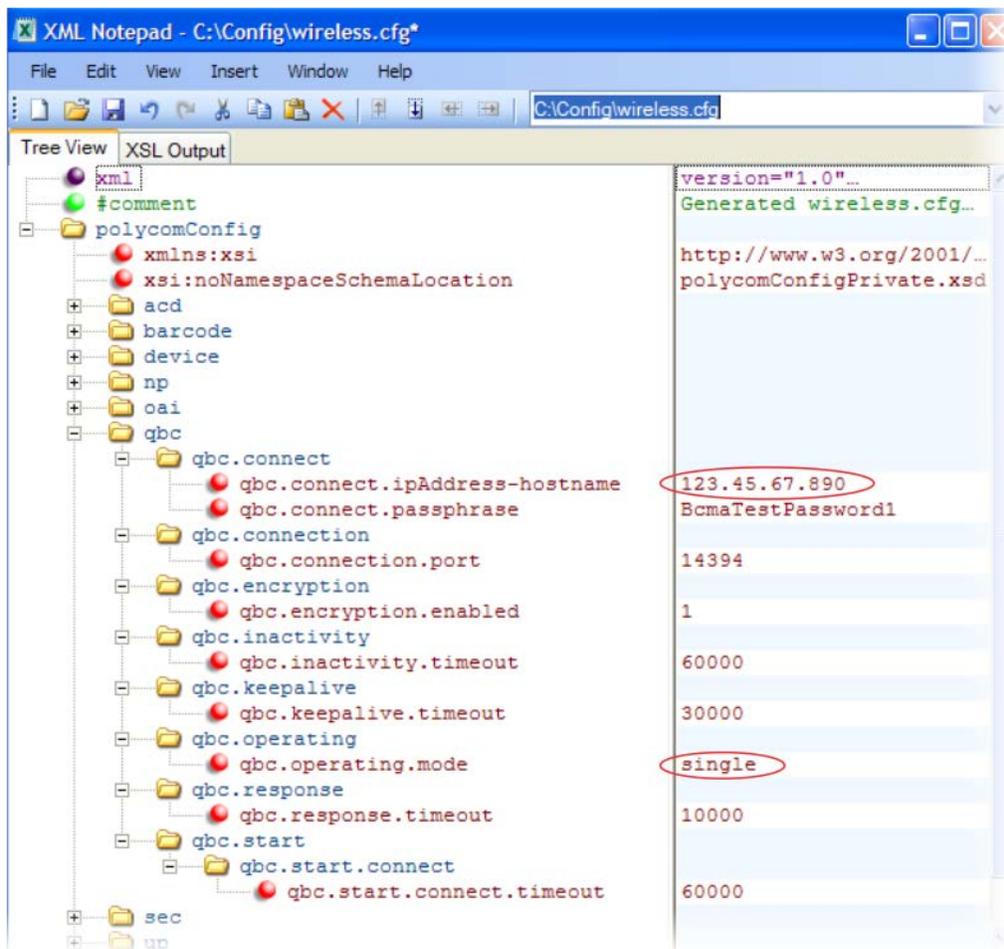
| Central Provisioning Server | template > parameter |
|--|---|
| In single endpoint mode, set the IP address or hostname of the computer running QBC | <code>wireless.cfg > qbc.connect.ipAddress-hostname</code> |
| To set the barcode scanner connector passphrase | <code>wireless.cfg > qbc.connect.passphrase</code> |
| Specify the port number used for connections from the handset | <code>wireless.cfg > qbc.connection.port</code> |
| Specify whether scanned data should be encrypted | <code>wireless.cfg > qbc.encryption.enabled</code> |
| In multiple endpoint mode, specify how long the barcode should wait before disconnecting from the computer | <code>wireless.cfg > qbc.inactivity.timeout</code> |
| Specify the QBC application operating mode | <code>wireless.cfg > qbc.operating.mode</code> |

Example QBC Configuration

The barcode configuration options that you set in `wireless.cfg` depend on whether you want to operate the QBC using single endpoint mode or multiple endpoint mode. The following example

shows the minimum configuration you need to set to begin using the QBC. If you want to operate the QBC with multiple endpoints, you can accept the default configuration. If you want to operate the QBC with a single endpoint, make the following changes in **wireless.cfg**:

- Locate the `qbc.connect.ipAddress-hostname` parameter and enter the IP address or hostname of the computer running QBC.
- Change the `qbc.operating.mode` parameter from `multi` to `single`.



Web Info: Installing and Configuring the QBC application.

For details on how to install and configure the Polycom QBC application, see the [Polycom Quick Barcode Connector Installation and Configuration Guide](#).

Configuring the Open Application Interface

Polycom’s Open Application Interface (OAI) enables you to use the SpectraLink handsets to retrieve and respond to information on third-party computer applications. To configure OAI, see [Table 7-44: Configuring the Open Application Interface \(OAI\)](#).



Web Info: Using the SpectraLink 8000 OAI Gateway

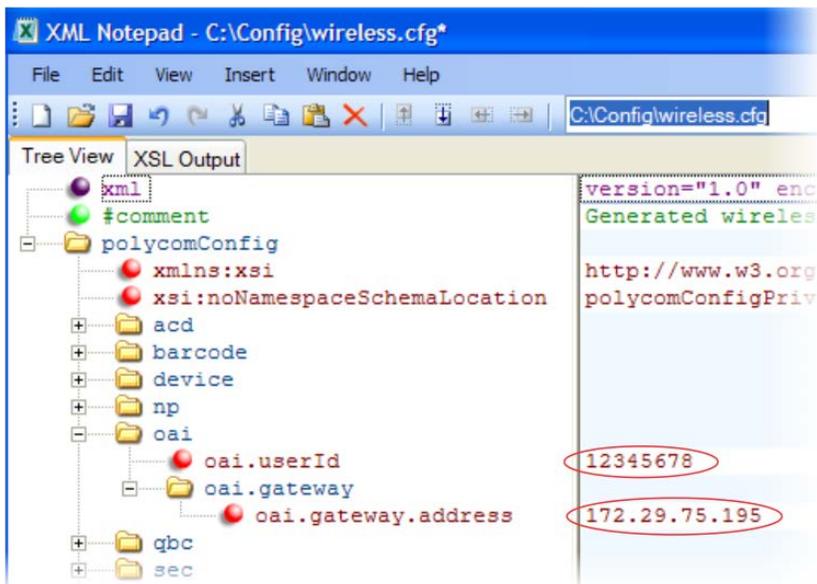
OAI v2.2 is supported by the SpectraLink handsets. For more information, see the [Polycom SpectraLink 8000 Open Applications Interface \(OAI\) Gateway Administration Guide](#).

Table 7-43: Configuring the Open Application Interface (OAI)

| | |
|---|--|
| Central Provisioning Server | template > parameter |
| Specify the lower four bytes of the six-byte OAI handset identifier in the OAI gateway..... | wireless.cfg > oai.userID |
| Specify the address of the OAI server | wireless.cfg > oai.gateway.address |

Example OAI Configuration

The following example shows the connection parameters you need to set for OAI communications with SpectraLink handsets. You will need to specify the OAI user ID and gateway address.



Enabling Location Services

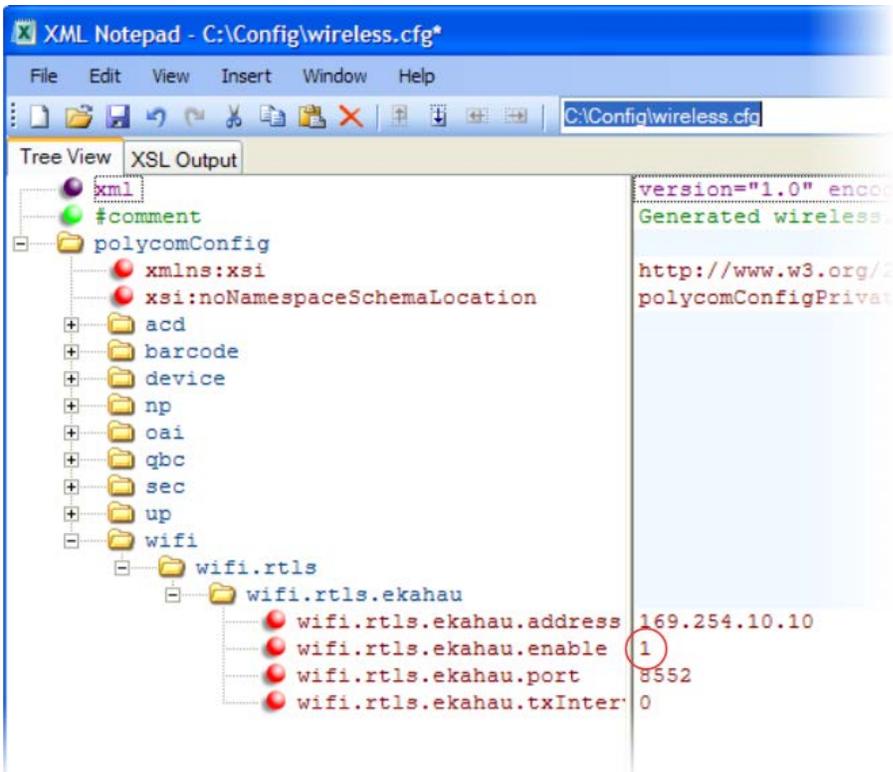
You can use location services to send reports for Ekahau® Real-Time Location Systems (RTLS) on the SpectraLink handsets. You can select a transmit interval and enter a static IP address for the Ekahau Positioning Engine™ (EPE) by configuring the parameters in [Table 7-45: Enabling Location Services](#). Location services are provided by the EPE 4.0 using Ekahau Location Protocol (ELP). For more information, see the [Ekahau Real Time Location System](#).

Table 7-44: Enabling Location Services

| Central Provisioning Server | template > parameter |
|--|--|
| Specify the IP address of the Ekahau Positioning Engine..... | wireless.cfg > wifi.rtls.ekahau.address |
| Enable or disable support for RTLS..... | wireless.cfg > wifi.rtls.ekahau.enable |
| Specify the port number of the Ekahau Positioning Engine | wireless.cfg > wifi.rtls.ekahau.port |
| Specify the maximum time between transmit intervals..... | wireless.cfg > wifi.rtls.ekahau.txInterval |

Example Location Service Integration Configuration

To use RTLS, enable the `wifi.rtls.ekahau.enable` parameter, as shown next. All other Ekahau parameter vales shown in the following example are the default values.



Chapter 8: Setting Up Phone Audio Features

After you set up your Polycom® phones on the network, phone users can send and receive calls using the default configuration. However, you might consider modifications that optimize the audio quality of your network.

Frequency bandwidth is one of the most critical elements affecting the intelligibility of speech in telephony. The frequency range that the human ear is most sensitive to is far beyond the capabilities of the plain old telephony system (POTS). In fact 80 percent of the frequencies in which speech occurs are not even used by public telephone networks because they only operate from 300Hz to 3.5 kHz. Complicating the intelligibility of telephony speech in today's world is background noise, variations in environmental reverberation, and communication among persons speaking a variety of native languages. While VoIP technology can broaden the frequency bandwidth and improve sound quality and intelligibility, it can also increase the network load and create a demand for lower raw bit rates. As [Table 8-7: Audio Codec Specifications](#) shows, Polycom offers phones with a range of codecs, including codecs with high frequency bandwidth and low raw bit rates.

This chapter describes the audio sound quality features and options you can configure for your Polycom phones. Use these features and options to optimize the conditions of your organization's phone network system.

This chapter shows you how to update your configuration for the following audio-related features:

- [Customizing Audio Sound Effects](#) Enables you to customize sound effects associated with incoming calls and other events.
- [Context Sensitive Volume Control](#) Choose the volume levels for the various audio outputs on the phone.
- [Voice Activity Detection](#) Conserves network bandwidth by detecting periods of relative 'silence' in the transmit data path and replacing that silence with special packets that indicate silence is occurring.
- [Generating Dual Tone Multi-Frequency \(DTMF\) Tones](#) Generates dual tone multi-frequency (DTMF) tones in response to user dialing on the dial pad.
- [DTMF Event RTP Payload](#) Conforms to RFC 2833, which describes a standard RTP-compatible technique for conveying DTMF dialing and other telephony events over an RTP media stream.
- [Acoustic Echo Cancellation](#) Employs advanced acoustic echo cancellation for handsfree operation.
- [Audio Codecs](#) Enables access to a wide range of industry standard audio codecs.

- [IP Type-of-Service](#) Enables the setting packet priority.
- [IEEE 802.1p/Q](#) The phone may tag all Ethernet packets it transmits with an 802.1Q VLAN header.
- [Voice Quality Monitoring](#) Generates various quality metrics including MOS and R-factor for listening and conversational quality. This feature is part of the Productivity Suite
- [Audible Ringer Location](#) Choose how to play out audio tones.
- [Notification Profiles](#) Define how your handset alerts you to phone events like incoming calls, instant messages, and pages. This feature is only available on SpectraLink handsets.
- [Bluetooth Headset Support](#) Enable a Bluetooth headset for use with the SpectraLink phones.

This chapter also outlines the following built-in audio processing features, which do not require any configuration changes to work:

- [Automatic Gain Control](#) Designed for handsfree operation, this feature boosts the transmit gain of the local user in certain circumstances.
- [Background Noise Suppression](#) Designed primarily for handsfree operation, this feature reduces background noise to enhance communication in noisy environments.
- [Comfort Noise Fill](#) Provides a consistent noise level to the remote user of a handsfree call.
- [Dynamic Noise Reduction](#) Provides maximum microphone sensitivity, while automatically reducing background noise. All Polycom phones automatically support this non-adjustable feature. This feature is also known as Noise Suppression.
- [Jitter Buffer and Packet Error Concealment](#) Employs a high-performance jitter buffer and packet error concealment system designed to mitigate packet inter-arrival jitter, and out-of-order, lost, or delayed packets.
- [Low-Delay Audio Packet Transmission](#) Minimizes latency for audio packet transmission.

To troubleshoot any problems with your Polycom phones on the network, see [Troubleshooting Your Polycom Phones](#). For more information on the configuration files, see [Polycom UC Software Configuration Files](#). For more information on the Web Configuration Utility, see [Getting Started with the Polycom Web Configuration Utility](#). For instructions on how to read the feature descriptions in this section, see [Reading the Feature Parameter Tables](#).

Customizing Audio Sound Effects

You can customize the audio sound effects that are used for incoming calls and other alerts using synthesized tones or sampled audio files. You can replace the default sampled audio files with your own custom **.wav** audio file format. The phone supports the following **.wav** audio file formats:

- mono G.711 (13-bit dynamic range, 8-khz sample rate)

- mono L16/16000 (16-bit dynamic range, 16-kHz sample rate)
- mono L16/32000 (16-bit dynamic range, 32-kHz sample rate)
- mono L16/44100 (16-bit dynamic range, 44.1 kHz sample rate)
- mono L16/48000 (16-bit dynamic range, 48-kHz sample rate)



Note: Supported Audio Formats

The L16/32000 wav format is supported only on the SoundStation IP 6000 and VVX 500 and 1500 phones. The L16/44100 wav format is supported on only the VVX 1500 phones. The L16/48000 wav format is supported only on the and VVX phones.

Your custom sampled audio files must be available at the path or URL specified by `saf.x` in [Table 8-1](#) so the phone can download them. Include the name of the file and the `.wav` extension in the path.

Table 8-1: Customizing Audio Sound Effects

| Central Provisioning Server | template > parameter |
|---|-----------------------|
| Specify a path or URL for the phone to download a custom audio file..... | site.cfg > saf.x |
| Specify the name, type, and value for a custom sound effect..... | region.cfg > se.pat.* |
| Web Configuration Utility | |
| To add, play, or delete a custom audio file, navigate to Preferences > Ringtones and expand the Custom Audio Files menu. | |

Example Configuration

The following example configuration illustrates how to add a custom sound effect from a sampled audio file. In the example, the custom audio files *MyTone.wav* and *Chirp.wav* have been added as sound effects 12 and 13. The `welcome` sound has been customized to use the sampled audio file 13 (*Chirp.wav*) with the label *Birds*. Ringtone 19 is named *Whistle* and is configured to use sampled audio file 12 (*MyTone.wav*).



The following illustration shows the custom ring tone *Whistle* as it displays on the phone menu:



Context Sensitive Volume Control

The parameters shown in [Table 8-2: Context Sensitive Volume Control](#) enable you to 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. While transmit levels are fixed according to the TIA/EIA-810-A standard, you can adjust the receive volume. The receiving volume of the handset and headset resets after each call to comply with regulatory requirements. The hands free speakerphone volume level remains at the same level as the previous call.

Table 8-2: Context Sensitive Volume Control

| Central Provisioning Server | template > parameter |
|---|---|
| Specify if a Bluetooth headset should be used for every call (SpectraLink 8400 Series only)
..... | site.cfg > voice.volume.persist.bluetooth.headset |
| Specify if the volume level of the handset, headset, and speakerphone should reset after each call
..... | site.cfg > voice.volume.persist.* |

Voice Activity Detection

The purpose of voice activity detection (VAD) is to detect periods of silence in the transmit data path so the phone doesn't have to transmit unnecessary data packets for outgoing audio. This process conserves network bandwidth. The VAD parameters in [Table 8-3: Voice Activity Detection \(VAD\)](#) will help you set up this feature. 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. The phone generates CN packets — also known as Silence Insertion Descriptor (SID) frames — and also decodes CN packets, to efficiently regenerate a facsimile of the background noise at the remote end.

Table 8-3: Voice Activity Detection (VAD)

| Central Provisioning Server | template > parameter |
|--|---|
| Specify if G.729 Annex B should be signaled..... | site.cfg > voice.vad.signalAnnexB |
| Enable or disable voice activity detection..... | site.cfg > voice.vadEnable |
| Specify the threshold between active voices and background voices..... | site.cfg > voice.vadThresh |

Generating Dual Tone Multi-Frequency (DTMF) Tones

The phone generates dual tone multi-frequency (DTMF) tones in response to user dialing on the dial pad. The parameters in the following table will help you set up this feature. These tones, commonly referred to as *touch tones*, are transmitted in the real-time transport protocol (RTP) streams of connected calls. The 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 end point.

Table 8-4: Dual Tone Multi-Frequency (DTMF) Tone Generation

| Central Provisioning Server | template > parameter |
|---|---|
| Specify if DTMF tones should be played through the speakerphone | sip-interop.cfg
> tone.dtmf.chassis.masking |
| Specify the frequency level of DTMF digits | sip-interop.cfg > tone.dtmf.level |
| Specify how long the phone should wait between DTMF digits | sip-interop.cfg > tone.dtmf.offTime |
| Specify how long the phone should play each DTMF tone for | sip-interop.cfg > tone.dtmf.onTime |
| Enable or disable DTMF encoding in an RTP stream | sip-interop.cfg > tone.dtmf.viaRtp |

DTMF Event RTP Payload

The phone is compatible with *RFC 2833—RTP Payload for DTMF Digits, Telephony Tones, and Telephony Signals*. RFC 2833 describes a standard RTP-compatible technique for conveying DTMF dialing and other telephony events over an RTP media stream. 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. Use [Table 8-5: Dual Tone Multi-Frequency \(DTMF\) Event RTP Payload](#) to set up this feature.

Table 8-5: Dual Tone Multi-Frequency (DTMF) Event RTP Payload

| Central Provisioning Server | template > parameter |
|--|---|
| Specify if the phone will use RFC 2833 to encode DTMF | sip-interop.cfg > tone.dtmf.rfc2833Control |
| Specify the phone-event payload encoding in the dynamic range to be used in SDP offers | sip-interop.cfg > tone.dtmf.rfc2833Payload |

Acoustic Echo Cancellation

Your Polycom phone uses advanced acoustic echo cancellation (AEC) for handsfree operation using the speakerphone. See [Table 8-6: Audio Codec Priority](#) for a list of audio codecs available for each phone and their priority. The phone also supports headset echo cancellation. The phones use both linear and non-linear techniques to aggressively reduce echo while permitting natural, full-duplex communication patterns.



Caution: Contact Polycom Support Before Modifying Acoustic Echo Cancellation Parameters

Consult Polycom customer support before you make changes to any acoustic echo cancellation parameters.

Audio Codecs

The following table details the audio codec support and priority for Polycom phones:

Table 8-6: Audio Codec Priority

| <i>Phone</i> | <i>Supported Audio Codecs</i> | <i>Priority</i> |
|---|---|-----------------|
| SoundPoint IP 321 and 331 | G.711m-law | 6 |
| | G.711a-law | 7 |
| | G.729AB | 8 |
| | iLBC (13.33kbps, 15.2kbps) | 0, 0 |
| SoundPoint IP 335, 450, 550, 560, and 650 | G.711m-law | 6 |
| | G.711a-law | 7 |
| | G.722 | 4 |
| | G.729AB | 8 |
| | iLBC (13.33kbps, 15.2kbps)
<i>Note:</i> When iLBC is used, only three-way conferencing is supported. | 0, 0 |
| SoundStation IP 5000 | G.711m-law | 6 |
| | G.711a-law | 7 |
| | G.722 | 4 |
| | G.729AB | 8 |
| | iLBC (13.33kbps, 15.2kbps)
<i>Note:</i> Only one of iLBC or G.729AB is supported. Selecting iLBC will cause the phone to reboot. | 0, 0 |
| SoundStation IP 6000 | G.711m-law | 6 |
| | G.711a-law | 7 |
| | G.722 | 4 |

| <i>Phone</i> | <i>Supported Audio Codecs</i> | <i>Priority</i> |
|-------------------------------|---|-----------------|
| | G.722.1 (32kbps) | 5 |
| | G.722.1C (48kbps) | 2 |
| | G.729AB | 8 |
| | iLBC (13.33kbps, 15.2kbps) | 0 |
| SoundStation Duo | G.711m-law | 6 |
| | G.711a-law | 7 |
| | G.722 | 4 |
| | G.729AB | 8 |
| | iLBC (13.33kbps, 15.2kbps) | 0, 0 |
| | Note: Only one of iLBC or G.729AB is supported. Selecting iLBC will cause the phone to reboot. | |
| VVX 500 and 1500 | G.711m-law | 6 |
| | G.711a-law | 7 |
| | G.719 (64kbps) | 0 |
| | G.722 | 4 |
| | G.722.1 (32kbps) | 5 |
| | G.722.1C (48kbps) | 2 |
| | G.729AB | 8 |
| | Siren14 (48kbps) | 3 |
| | iLBC (13.33kbps, 15.2kbps) | 0, 0 |
| SoundStructure VoIP Interface | G.711m-law | 6 |
| | G.711a-law | 7 |
| | G.729AB | 8 |
| | G.722 | 4 |
| | G.722.1 (32kbps) | 5 |
| | G.722.1C (48kbps) | 2 |

| <i>Phone</i> | <i>Supported Audio Codecs</i> | <i>Priority</i> |
|-----------------------|-------------------------------|-----------------|
| SpectralLink Handsets | G.711m-law | 6 |
| | G.711a-law | 7 |
| | G.722 | 4 |
| | G.722.1 (32kbps) | 5 |
| | G.729AB | 8 |

The following [Table 8-7: Audio Codec Specifications](#) summarizes the audio codecs supported on Polycom phones:

Table 8-7: Audio Codec Specifications

| <i>Algorithm</i> | <i>Reference</i> | <i>Raw Bit Rate</i> | <i>IP Bit Rate</i> | <i>Sample Rate</i> | <i>Default Payload Size</i> | <i>Effective Audio Bandwidth</i> |
|------------------|------------------|---------------------|--------------------|--------------------|-----------------------------|----------------------------------|
| G.711 u-law | RFC 1890 | 64 Kbps | 80 Kbps | 8 Ksps | 20 ms | 3.5 KHz |
| G.711 a-law | RFC 1890 | 64 Kbps | 80 Kbps | 8 Ksps | 20 ms | 3.5 KHz |
| G.719 | RFC 5404 | 32 Kbps | 48 Kbps | 48 Ksps | 20 ms | 20 KHz |
| | | 48 Kbps | 64 Kbps | | | |
| | | 64 Kbps | 80 Kbps | | | |
| G.711 | RFC 1890 | 64 Kbps | 80 Kbps | 16 Ksps | 20 ms | 7 KHz |
| G.722.1 | RFC 3047 | 16 Kbps | 32 Kbps | 16 Ksps | 20 ms | 7 KHz |
| | | 24 Kbps | 40 Kbps | | | |
| | | 32 Kbps | 48 Kbps | | | |
| G.722.1C | G7221C | 224 Kbps | 40 Kbps | 32 Ksps | 20 ms | 14 KHz |
| | | 32 Kbps | 48 Kbps | | | |
| | | 48 Kbps | 64 Kbps | | | |
| G.729AB | RFC 1890 | 8 Kbps | 24 Kbps | 8 Ksps | 20 ms | 3.5 KHz |
| Lin16 | RFC 1890 | 128 Kbps | 132 Kbps | 8 Ksps | 10 ms | 3.5 KHz |
| | | 256 Kbps | 260 Kbps | 16 Ksps | | 7 KHz |
| | | 512 Kbps | 516 Kbps | 32 Ksps | | 14 KHz |
| | | 705.6 Kbps | 709.6 Kbps | 44.1 Ksps | | 20 KHz |
| | | 768 Kbps | 772 Kbps | 48 Ksps | | 22 KHz |

| Algorithm | Reference | Raw Bit Rate | IP Bit Rate | Sample Rate | Default Payload Size | Effective Audio Bandwidth |
|-----------|-----------|--------------|-------------|-------------|----------------------|---------------------------|
| Siren14 | SIREN14 | 24 Kbps | 40 Kbps | 32 Ksps | 20 ms | 14 KHz |
| | | 32 Kbps | 48 Kbps | | | |
| | | 48 Kbps | 64 Kbps | | | |
| Siren22 | SIREN22 | 32 Kbps | 48 Kbps | 48 Ksps | 20 ms | 22 KHz |
| | | 48 Kbps | 64 Kbps | | | |
| | | 64 Kbps | 80 Kbps | | | |
| iLBC | RFC 3951 | 13.33 Kbps | 31.2 Kbps | 8 Ksps | 30 ms | 3.5 KHz |
| | | 15.2 Kbps | 24 Kbps | | 20 ms | |



Note: Network Bandwidth Requirements for Encoded Voice

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 48 kbps for both the receive and transmit signals consumes about 100 kbps of network bandwidth (two-way audio).

Use [Table 8-8: Audio Codec Priorities](#) to specify the priority for audio codecs on your Polycom phones.

Table 8-8: Audio Codec Priorities

| | |
|--|--------------------------------|
| Central Provisioning Server | template > parameter |
| To specify the priority for a codec site.cfg > voice.codecPref.<nameOfCodec> | |
| Web Configuration Utility | |
| To enable or disable codecs and specify codec priority, navigate to Settings > Codec Profiles and expand the Audio Priority menu. | |

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. See [Table 8-9: IP Type-of-Service \(ToS\)](#) for available parameters. 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.

Table 8-9: IP Type-of-Service (ToS)

| Central Provisioning Server | template > parameter |
|--|---------------------------------|
| Set the IP header bits for call control | template > qos.ip.callControl.* |
| Set the IP header bits for RTP | template > qos.ip.rtp.* |
| Set the IP header bits for RTP video | template > qos.ip.rtp.video.* |
| Web Configuration Utility | |
| Set the QoS IP settings by navigating to Settings > Network > QoS . | |

IEEE 802.1p/Q

The phone will tag all Ethernet packets it transmits with an 802.1Q VLAN header when:

- 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 LLDP (see [DHCP Menu](#)).

Use [Table 8-10: IEEE 802.1p/Q](#) to set values. The 802.1p/Q `user_priority` field can be set to a value from 0 to 7. The `user_priority` can be configured specifically for RTP packets and call control packets, such as SIP signaling packets, with default settings configurable for all other packets.

Table 8-10: IEEE 802.1p/Q

| Central Provisioning Server | template > parameter |
|--|---|
| Set the user priority for packets without a per-packet protocol setting (including 802.1p/Q | site.cfg > qos.ethernet.other.user_priority |
| Web Configuration Utility | |
| To set the user priority for 802.1p/Q packets, navigate to Settings > Network > QoS and expand the Other Protocols menu. | |

Voice Quality Monitoring

Certain phones can be configured to generate various quality metrics for listening quality and conversational quality. These metrics can be sent between the phones in RTCP XR packets. The metrics can also be sent as SIP PUBLISH messages to a central voice quality report collector.



Web Info: RTCP XR Packet Compliancy

The RTCP XR packets are compliant with [RFC 3611—RTP Control Extended Reports \(RTCP XR\)](#). The packets are sent to a report collector as specified in draft RFC [draft-ietf-sipping-rtcp-summary-02](#).

The collection of these metrics is supported on the SoundPoint IP 321/331/335, 450, 550, 560, and 650 phones, SoundStation IP 5000 conference phones, the VVX 500 and 1500 phones, and SpectraLink handsets.



Note: Activating the Voice Quality Monitoring Feature

This feature requires a license key for activation on all phones except the VVX 500 and 1500 phones and the SpectraLink handsets. For more information, contact your Certified Polycom Reseller.

Three types of quality reports can be enabled:

- **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.

A wide range of performance metrics are generated, the parameters for which are shown in [Table 8-11: Voice Quality Monitoring \(VQM\)](#). 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 computed using other metrics as input, such as listening Mean Opinion Score (MOS), conversational MOS, listening R-factor, and conversational R-factor.

Table 8-11: Voice Quality Monitoring (VQM)

| Central Provisioning Server | template > parameter |
|--|--|
| Specify the warning threshold for alerts | features.cfg > voice.qualityMonitoring.collector.alert.* |
| Enable the generation of quality reports | features.cfg > voice.qualityMonitoring.collector.enable.* |
| Specify the server address and port | features.cfg > voice.qualityMonitoring.collector.server.x.* |
| Enable the generation of RTCP-XR packets | features.cfg > voice.qualityMonitoring.rtcpxr.enable |

Audible Ringer Location

You can choose where all audio alerts, including incoming call alerts, are played out on all SoundPoint IP and VVX phones installed with UC Software 3.3.0 or later. Use [Table 8-13: Audible Ringer Location](#) to specify where you hear audio. You can specify the audio to play from the handsfree speakerphone (default), the handset, the headset, or the active location. If you choose the active location, audio alerts will be played out through the handset or headset if they are off hook. Otherwise, alerts play through the speakerphone.

Table 8-12: Audible Ringer Location

| Central Provisioning Server | template > parameter |
|--|-----------------------------------|
| Specify where audio alerts play out from..... | reg-advanced.cfg > se.destination |
| Local Phone User Interface | |
| Specify where incoming call ringing plays out from the Audible Ringer menu, accessible from Menu > Settings > Basic > Preferences > Audible Ringer . | |

Notification Profiles

The SpectraLink handsets support four profiles for notification alerts: **Normal**, **Silent**, **Meeting**, and **Custom1**. You can customize each profile with unique ringtones, alerts, and vibrations for specific situations. For example, you can customize barcode scan alerts or when you receive an instant message. See [Table 8-14: Notification Profiles](#) for a list of available parameters.

By default, the ringing and alert volumes are at the same level. You can configure the ringer volume for ringing only and set a distinct alert volume for each alert type. By default, the phone will maintain changes you make to the ringer volume when the phone reboots or restarts.

Table 8-13: Notification Profiles

| Central Provisioning Server | template > parameter |
|---|---------------------------------------|
| Specify the initial notification profile, this can be overridden from the handset | wireless.cfg > np.selected |
| Specify the label, for the custom1, meeting, normal, and silent profiles | wireless.cfg > np.<profile>.label |
| Customize alerts for the custom1, meeting, normal, and silent profiles | wireless.cfg > np.<profile>.alert.* |
| Customize ringing for the custom1, meeting, normal, and silent profiles | wireless.cfg > np.<profile>.ringing.* |

Local Phone User Interface

To configure a notification profile on the phone, navigate to **Settings > Basic Settings > Notification Profiles** and select a notification profile to edit.

Bluetooth Headset Support

You can use Bluetooth v2.1 headsets with your SpectraLink handsets. To use a Bluetooth headset, you need to enable the Bluetooth headset feature and turn on the Bluetooth radio, as shown in [Table 8-15: Bluetooth Headset Support](#).



Troubleshooting: Using a Bluetooth Headset Affects my Phone's Voice Quality

You may not experience the highest voice quality if you use a Bluetooth headset while the 2.4 GHz band is enabled or while you are in an environment with many other Bluetooth devices or other 2.4 GHz wireless devices. This possible loss in voice quality is due to inherent limitations with Bluetooth technology.

Table 8-14: Bluetooth Headset Support

| Central Provisioning Server | template > parameter |
|---|---|
| To enable or disable the Bluetooth headset feature | features.cfg > feature.bluetooth.enabled |
| To turn the Bluetooth radio (transmitter/receiver) on or off..... | features.cfg > bluetooth.radioOn |

Built-In Audio Processing Features

Your Polycom phone has the following built-in audio processing features: automatic gain control, background noise suppression, comfort noise fill, dynamic noise reduction, jitter buffer and packet error concealment, and low delay audio packet transmission. These features work automatically, without configuration changes.

Automatic Gain Control

Automatic Gain Control (AGC) is applicable to handsfree operation and is used to boost the transmit gain of the local talker in certain circumstances. This increases the effective user-phone radius and helps with the intelligibility of soft-talkers.

Background Noise Suppression

Background noise suppression (BNS) is designed primarily for handsfree operation and reduces background noise to enhance communication in noisy environments.

Comfort Noise Fill

Comfort noise fill is designed to help provide a consistent noise level to the remote user of a handsfree call. Fluctuations in perceived background noise levels are an undesirable side effect of the non-linear component of most AEC systems. This feature uses noise synthesis techniques to smooth out the noise level in the direction toward the remote user, providing a more natural call experience.

Dynamic Noise Reduction

Dynamic noise reduction (DNR) provides maximum microphone sensitivity, while automatically reducing background noise— from fans, projectors, heating and air conditioning—for clearer sound and more efficient conferencing.

Jitter Buffer and Packet Error Concealment

The phone employs 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.

Low-Delay Audio Packet Transmission

The phone is designed to minimize latency for audio packet transmission.

Chapter 9: Setting Up Phone Video Features

After you set up the Polycom® phones on your network, you can allow users to place and answer calls using the default configuration. However, you may require some video-related changes to optimize your system for best results.

The Polycom VVX® 1500 Business Media Phone supports transmission and reception of high quality video images. The video is compatible with RFC 3984 - RTP Payload Format for H.264 Video, RFC 4629 - RTP Payload Format for ITU-T Rec. H.263 Video, and RFC 5168 - XML Schema for Media Control.

This chapter shows you how to update your configuration for the following video-related features:

- [Video Transmission](#) Start or stop the transmission of video on the VVX 1500.
- [Video Codecs](#) Support for industry standard video codecs (on the VVX 1500 phones only).
- [H.323 Protocol](#) Support for the H.323 protocol (for the VVX 1500 phones only).
- [Switching Between Voice and Video During Calls](#) Enable VVX 1500 phones to switch between voice only and video only calls.

To troubleshoot any problems with your Polycom phones on the network, see [Troubleshooting Your Polycom Phones](#). For more information on the configuration files, see [Polycom UC Software Configuration Files](#). For more information on the Web Configuration Utility, see [Getting Started with the Polycom Web Configuration Utility](#). For instructions on how to read the feature descriptions in this section, see [Reading the Feature Parameter Tables](#).

Video Transmission

By default, at the start of a video call, the VVX 1500 phone transmits an RTP encapsulated video stream with images captured from the local camera. Users can stop and start video transmission by pressing the **Video** key, and then selecting the **Stop** or **Start** soft key.

You can configure:

- [Video Transmission Parameters](#)
- [Video and Camera View Parameters](#)
- [Video Camera Parameters](#)

You can use the parameters in [Table 9-1: Video Transmission Parameters](#) to configure video transmission on your VVX 1500 phone:

Table 9-1: Video Transmission Parameters

| Central Provisioning Server | template > parameter |
|---|--|
| Specify if video calls should use a full screen layout..... | video.cfg > video.autoFullScreen |
| Specify when video transmission should start in a call..... | video.cfg > video.autoStartVideoTx |
| Set the call rate for a video call (can be changed on the phone) | video.cfg > video.callRate |
| Specify whether the phone is forced to send RTCP feedback messages to request fast update I-frames for video calls | video.cfg > video.forceRtcpVideoCodecControl |
| Set the maximum call rate for a video call (the maximum rate set from the phone cannot exceed this) | video.cfg > video.maxCallRate |
| Specify the quality of video to be shown in a call or conference | video.cfg > video.quality |
| <hr/> | |
| Web Configuration Utility | |
| To configure video processing options, navigate to Preferences > Video Processing and expand the General menu. | |
| <hr/> | |
| Local Phone User Interface | |
| To configure video processing options, navigate to Menu > Settings > Basic > Video > Video Call Settings . | |

You can use the parameters in [Table 9-2: Video and Camera View Parameters](#) to set the video and local camera view settings on your VVX 1500 phone:

Table 9-2: Video and Camera View Parameters

| Central Provisioning Server | template > parameter |
|--|--|
| Specify the view of the video window in normal viewing mode | video.cfg > video.screenMode |
| Specify the view of the video window in full screen viewing mode..... | video.cfg > video.screenModeFS |
| Specify if the local camera view is shown in the full screen layout | video.cfg > video.localCameraView.fullscreen.enabled |
| Determine how the local camera view is shown | video.cfg > video.localCameraView.fullscreen.mode |
| <hr/> | |
| Web Configuration Utility | |
| To configure the video and camera view settings, navigate to Preferences > Video Processing . To configure the Screen Mode, expand the General menu. To configure the camera view, expand the Local Camera View Settings menu. | |
| <hr/> | |
| Local Phone User Interface | |
| To configure the video and camera view settings, navigate to Menu > Settings > Basic > Video and configure Video Screen Mode and Local Camera View . | |

You can use the parameters in [Table 9-3: Video Camera Parameters](#) to configure the video camera on your VVX 1500 phone:

Table 9-3: Video Camera Parameters

| Central Provisioning Server | template > parameter |
|--|---|
| Set the brightness level..... | video.cfg > video.camera.brightness |
| Set the contrast level..... | video.cfg > video.camera.contrast |
| Specify if flicker avoidance is automatic, suited for Europe/Asia, or North America
..... | video.cfg > video.camera.flickerAvoidance |
| Set the frame rate | video.cfg > video.camera.frameRate |
| Set the saturation level | video.cfg > video.camera.saturation |
| Set the sharpness level..... | video.cfg > video.camera.sharpness |
| Web Configuration Utility | |
| To set the video camera settings, navigate to Preferences > Video Processing and expand the Camera Settings menu. | |
| Local Phone User Interface | |
| To set the video camera settings, navigate to Menu > Settings > Basic > Video > Camera Settings . | |

Video Codecs

See [Table 9-4: Video Codec Specifications](#) for a summary of the VVX 1500 phone's video codec support:

Table 9-4: Video Codec Specifications

| <i>Algorithm</i> | <i>MIME Type</i> | <i>Frame Size</i> | <i>Bit Rate (kbps)</i> | <i>Frame Rate (fps)</i> |
|------------------|------------------------------------|---|------------------------|-------------------------|
| H.261 | H261/90000 | Tx Frame size: CIF, QCIF,
SQCIF
RX Frame size: CIF, QCIF | 64 to 768 | 5 to 30 |
| H.263 | H263/90000,
H263-
1998/90000 | Tx Frame size: CIF, QCIF
Rx Frame size: CIF, QCIF,
SQCIF, QVGA, SVGA, SIF | 64 to 768 kbps | 5 to 30 |
| H.264 | H264/90000 | Tx Frame size: CIF, QCIF
Rx Frame size: CIF, QCIF,
SQCIF, QVGA, SVGA, SIF | 64 to 768 | 5 to 30 |

You can configure the parameters in [Table 8-5: Video Codec Parameters](#) to prioritize and adjust the video codecs that your VVX 1500 phone uses:

Table 9-5: Video Codec Parameters

| Central Provisioning Server | template > parameter |
|---|---|
| Prioritize the video codecs from 1 to 4..... | video.cfg > video.codecPref.* |
| Adjust the parameters for the H261, H263, H2631998, and H264 codec profiles
..... | video.cfg > video.profile.<codec>.* |
| Web Configuration Utility | |
| To set the priority for the video codecs, navigate to Settings > Codec Priorities , expand the Video Priority menu, and use the arrow keys to re-order the codecs. | |
| To adjust the parameters for the video codecs, navigate to Settings > Codec Profile > Video . | |

H.323 Protocol

As of SIP 3.2.2, the VVX 1500 phone supports telephony signaling via the H.323 protocols. This protocol enables direct communication with H.323 endpoints, gatekeepers, call servers, media servers, and signaling gateways.



Note: Activating H.323 Video

You will need a license key to activate H.323 video on your VVX 1500 phone; the license is installed on the VVX 1500D. For more information, contact your Certified Polycom Channel Partner.

The VVX 1500 can support SIP and H.323 signaling simultaneously, and you can bridge both types of calls during multi-party conference calls. The phone can automatically detect the correct or optimal signaling protocol when dialing a call from the contact directory or the corporate directory. While SIP supports server redundancy and several transport options, only a single configured H.323 gatekeeper address per phone is supported. The phone does not require H.323 gatekeepers, but if H.323 gatekeepers are available, they will be used. If an H.323 gatekeeper is not configured or is unavailable, you can still enable your phone to make H.323 calls.

Support of the SIP protocol for telephony signaling can be disabled on the VVX 1500 such that all calls will be routed via the H.323 protocol.

This section provides detailed information on:

- [Supported Video Standards](#)
- [Supported Polycom Interoperability](#)
- [Using the H.323 Protocol](#)

For a list of all H.323 parameters, see [Table 9-6: H.323 Protocol Parameters](#), shown next.

Table 9-6: H.323 Protocol Parameters

| Central Provisioning Server | template > parameter |
|---|---|
| Specify if the user is presented with protocol routing choices
..... | reg-advanced.cfg and site.cfg > up.manualProtocolRouting |
| Set soft keys for protocol routing | reg-advanced.cfg and site.cfg > up.manualProtocolRouting.softKeys |
| Enable or disable auto-answer for all H.323 calls..... | reg-advanced.cfg and h323.cfg
> call.autoAnswer.H323 |
| Specify if the phone can make calls using H.323 even if an H.323 gatekeeper is not configured or is unavailable..... | sip-interop.cfg > call.enableOnNotRegistered |
| Specify if video should begin immediately after a call is auto-answered
..... | reg-advanced.cfg > call.autoAnswer.videoMute |
| Specify whether SIP or H.323 is the preferred call protocol..... | video.cfg
> call.autoRouting.preferredProtocol |
| Specify if calls should be routed by line or by protocol.... | sip-interop.cfg > call.autoRouting.preference |
| Enable or disable H.323 signaling for the line registration..... | sip-interop.cfg > reg.x.protocol.H323 |
| Specify the H.323 server settings for a specific registration..... | site.cfg > reg.x.server.H323.* |
| Specify the H.323 protocol settings..... | h323.cfg > volpProt.H323.* |
| Specify the H.323 server settings..... | h323.cfg > volpProt.server.H323.* |
| Configure the H.323 media encryption parameters..... | site.cfg > sec.H235.mediaEncryption.* |
| <hr/> | |
| Web Configuration Utility | |
| To configure auto answer and protocol routing, navigate to Preferences > Additional Preferences and expand the Auto Answer and Protocol Routing menus. | |
| To specify the global H.323 settings, navigate to Settings > H.323 . | |
| To specify the H.323 settings for a specific registration, navigate to Settings > Lines , choose a line from the left pane, and expand the H.323 Settings menu. | |
| To specify the global H.323 Line Settings, navigate to Simple Setup and expand the H.323 Line Settings , H.323 Global Gatekeeper Settings , and H.323 Local Port Settings menus. | |
| <hr/> | |
| Local Phone User Interface | |
| To specify the global H.323 settings, navigate to Menu > Settings > Advanced > Call Server Configuration > H.323 . | |
| To specify the per-registration H.323 settings, navigate to Menu > Settings > Advanced > Line Configuration > Line X > H.323 Protocol . | |

Supported Video Standards

See [Table 9-7: Supported Video Standards](#) to find out which standards are supported by the H.323 feature.

Table 9-7: Supported Video Standards

| <i>Standard</i> | <i>Description</i> |
|---|--|
| ITU-T Recommendation H.323 (2003) | Packet-based multimedia communications systems |
| ITU-T Recommendation Q.931 (1998) | ISDN user-network interface layer 3 specification for basic call control |
| ITU-T Recommendation H.225.0 (2003) | Call signaling protocols and media stream packetization for packet-based multimedia communications systems |
| ITU-T Recommendation H.245 (5/2003) | Control protocol for multimedia communication |
| ITU-T Recommendation H.235.0 - H.235.9 (2005) | Security and encryption for H Series (H.323 and other H.245 based) multimedia terminals |
| ITU-T Recommendation H.350.1 (8/2003) | Directory services architecture for H.323 (through a Polycom CMA system only) |

Supported Polycom Interoperability

Video calls are supported by the Polycom endpoints/bridges/call servers (or gatekeepers)/media servers listed in [Table 9-8: Supported Polycom Interoperability](#).

Table 9-8: Supported Polycom Interoperability

| <i>Make/Model</i> | <i>Protocol</i> | <i>Software Version</i> |
|---|-----------------|-------------------------|
| Polycom CMA System | H.323 | SW 5.0 |
| Polycom HDX® 9000 series | SIP/ISDN/H.323 | SW 2.6.0 |
| Polycom HDX® 8000 series | SIP/ISDN/H.323 | SW 2.6.0 |
| Polycom HDX® 7000 series | SIP/ISDN/H.323 | SW 2.6.0 |
| Polycom HDX® 6000 | SIP/ISDN/H.323 | SW 2.6.0 |
| Polycom HDX® 4000 series | SIP/ISDN/H.323 | SW 2.6.0 |
| Polycom RMX® 2000 | H.323 | SW 4.0.2.7 |
| Polycom Quality Definition Experience™ (QDX™) | H.323 | SW 4.0, 4.0.1 |
| Polycom RMX® 1000 | H.323 | SW 1.1.1.8787 |
| Polycom RMX® 2000 | H.323 | SW 5.0.1.24, 6.0 |

| <i>Make/Model</i> | <i>Protocol</i> | <i>Software Version</i> |
|-----------------------------------|-----------------|-------------------------|
| Polycom RSS™ | H.323 | SW 6.0 |
| Polycom VBP™ 6400-ST series | H.323 | SW 9.1.5.1 |
| Polycom VBP™ 5300-ST series | H.323 | SW 9.1.5.1 |
| Polycom VBP™ 5300-E series | H.323 | SW 9.1.5.1 |
| Polycom VBP™ 4350 series | H.323 | SW 9.1.5.1 |
| Polycom VBP™ 200 | H.323 | SW 9.5.2 |
| Polycom VSX® 8000 | SIP/ISDN/H.323 | SW 9.0.6 |
| Polycom VSX® 7000s and VSX® 7000e | SIP/ISDN/H.323 | SW 9.0.6 |
| Polycom VSX® 6000 and 6000a | SIP/ISDN/H.323 | SW 9.0.5.1 |
| Polycom VSX® 5000 | SIP/ISDN/H.323 | SW 9.0.5.1 |
| Polycom VSX® 3000 | SIP/ISDN/H.323 | SW 9.0.5.1 |
| Polycom V700™ | SIP/ISDN/H.323 | SW 9.0.5.1 |
| Polycom V500™ | SIP/ISDN/H.323 | SW 9.0.5.1 |



Web Info: Viewing an Updated List of Polycom Video Support with Third Party Products

See the *UC Software Release Notes* on [Latest Polycom UC Software Release](#) for the latest list of supported Polycom endpoints/bridges/call servers (or gatekeepers)/media servers and any supported third party products. Any issues (and possible workarounds) with any of the above-mentioned products are also documented in the *Release Notes*.

Using the H.323 Protocol

The following information should be noted:

- If the phone has only the H.323 protocol enabled, it cannot be used to answer SIP calls.
- If the phone has only the SIP protocol enabled, it cannot be used to answer H.323 calls.
- If both SIP and H.323 protocols are disabled by mistake, the phone will continue to work as a SIP-only phone; however, the phone is not registered (you are able to send and receive SIP URL calls).
- The phone will store the protocol used to place a call in the placed call list.
- The protocol to be used when placing a call from the user's local contact directory is unspecified by default. The user can select SIP or H.323.

- The protocol that is used when placing a call from the user's corporate directory depends on the order of the attributes in the corporate directory. If only `SIP_address` is defined, then the SIP protocol is used. If only `H323_address` is defined, then the H.323 protocol is used. If both are defined, then the one that is defined first is used. For example, if `dir.corp.attribute.4.type` is `SIP_address` and `dir.corp.attribute.5.type` is `H323_address`, then the SIP protocol is used.
- By default, when more than one protocol is available, each protocol displays as a soft key and the user can choose which protocol to use.
- Calls made using H.323 cannot be forwarded or transferred.
 - The **Transfer** and **Forward** soft keys are not displayed during an H.323 call on a VVX 1500 phone. The **Forward** soft key is not displayed on the idle display of a VVX 1500 phone if the primary line is an H.323 line.
 - If a VVX 1500 user presses the **Transfer** soft key during an H.323 call on a VVX 1500 phone, no action is taken.
 - The auto-divert field in the local contact directory entry is ignored when a call is placed to that contact using H.323.
 - If a conference host ends a three-way conference call and one of the parties is connected by H.323, that party is not transferred to the other party that was part of the conference call.

The next graphic shows an example of a **sip-h323.cfg** file and the parameters you will need to configure:

- To configure both SIP and H.323 protocols.
- To set up a SIP and H.323 dial plan—Numbers with the format `0xxx` are placed on a SIP line and numbers with the format `33xx` are placed on an H.323 line.
- To set up manual protocol routing using soft keys—If the protocol to use to place a call cannot be determined, the **Use SIP** and **Use H.323** soft keys appear, and the user must select one for the call to be placed.
- To configure auto-answering on H.323 calls only.
- To set the preferred protocol to SIP.
- To set to configure one SIP line, one H.323 line, and a dual protocol line—both SIP and H.323 can be used.
- To set the preferred protocol for off-hook calls on the third (dual protocol) line to SIP.

| | |
|------------------------------------|-----------------------|
| phone | |
| voIpProt | |
| SIP | |
| voIpProt.SIP.enable | 1 |
| H323 | |
| voIpProt.H323.enable | 1 |
| dialplan | |
| digitmap | |
| dialplan.digitmap | 0xxxS 33xxH |
| user_preferences | |
| up.manualProtocolRouting | 1 |
| up.manualProtocolRouting.softKeys | 1 |
| call | |
| call.autoAnswer.SIP | 0 |
| call.autoAnswer.H323 | 1 |
| call.autoAnswer.micMute | 1 |
| call.autoAnswer.videoMute | 0 |
| call.autoRouting.preference | line |
| call.autoRouting.preferredProtocol | SIP |
| call.autoOffHook.3.protocol | SIP |
| reg | |
| reg.1.address | 1301 |
| reg.1.server.1.address | sipserver.polycom.com |
| reg.1.protocol.SIP | 1 |
| reg.1.protocol.H323 | 0 |
| reg.1.label | 1301S |
| reg.2.address | 1302 |
| reg.2.server.1.address | 172.88.2.123 |
| reg.2.protocol.SIP | 0 |
| reg.2.protocol.H323 | 1 |
| reg.2.label | 1302H |
| reg.3.address | 1303 |
| reg.3.server.1.address | sipserver.polycom.com |
| reg.3.server.2.address | 172.88.2.123 |
| reg.3.protocol.SIP | 1 |
| reg.3.protocol.H323 | 1 |
| reg.3.label | 1303D |

Switching Between Voice and Video During Calls

You can enable VVX 1500 phones to switch between voice and video during calls. Use [Table 9-9: Voice and Video Toggle Parameters](#) to locate the available parameters. If this feature is enabled, users can switch between audio-only calls, and calls with audio and video. Users can make audio calls by default, and select a **Voice/Video** if they want to add video to the call. Once a video call has ended, the phone will switch back to audio-only.

Table 9-9: Voice and Video Toggle Parameters

| Central Provisioning Server | template > parameter |
|--|---|
| Enable or disable the audio/video toggle feature..... | features.cfg > feature.audioVideoToggle.enabled |
| Allow the user to select the call mode to use when using SIP protocol only | video.cfg > video.callMode.default |

Chapter 10: Setting Up User and Phone Security Features

After setting up your Polycom® phones on your network, users can place and answer calls using the default configuration. However, you may require some security-related changes to optimize your system for best results.

This chapter shows you how to update your configuration for the following security-related features:

- [Local User and Administrator Passwords](#) Several local settings menus are protected with two privilege levels—user and administrator—each with its own password.
- [Incoming Signaling Validation](#) Levels of security are provided for validating incoming network signaling.
- [Configuration File Encryption](#) Confidential information stored in configuration files can be protected (encrypted). The phone can recognize encrypted files, which it downloads from the provisioning server, and it can encrypt files before uploading them to the provisioning server.
- [Digital Certificates](#) Most Polycom phones support digital certificates and associated private keys.
- [Generating a Certificate Signing Request](#) Create a request to obtain a device certificate.
- [TLS Profiles](#) Configure your phone with a profile that specifies trusted digital certificates. You can also install and specify custom certificates.
- [Supporting Mutual TLS Authentication](#) Support phone authentication of the server and server authentication of the phone.
- [Configurable TLS Cipher Suites](#) Control which of cipher suites will be offered/accepted during TLS session negotiation.
- [Secure Real-Time Transport Protocol](#) Encrypting audio streams to avoid interception and eavesdropping. Encrypting audio streams to avoid interception and eavesdropping.
- [Locking the Phone](#) Prevent access to the phone menu and to key presses.
- [Locking the Keypad on Your SpectraLink Handset](#) Lock the keypad on the SpectraLink handset.
- [Secondary Port Link Status Report](#) SoundPoint IP phones equipped with a secondary (PC) port can act as a pass-through switch for externally attached devices.
- [Supporting 802.1X Authentication](#) Authenticate devices connecting to a local area network (LAN) or a wireless local area network (WLAN).

- [Using User Profiles](#) Access your personal phone settings from any phone in your organization's network.

To troubleshoot any problems with your Polycom phones on the network, see [Troubleshooting Your Polycom Phones](#). For more information on the configuration files, see [Polycom UC Software Configuration Files](#). For more information on the Web Configuration Utility, see [Getting Started with the Polycom Web Configuration Utility](#). For instructions on how to read the feature descriptions in this section, see [Reading the Feature Parameter Tables](#).

Local User and Administrator Passwords

Several local settings menus are protected with user and administrator passwords. The phone will prompt you for a user or administrator password before you can access certain menu options. If the phone requires the administrator password, you may be able to use the user password, but you will be presented with limited menu options. If the phone prompts you for the user password, you may use the administrator password (you will see the same menus as the user). The Web Configuration Utility is protected by the user and administrator password and displays different features and options depending on which password you use. The default user password is **123** and the default administrator password is **456**. You should change the administrator password from the default value. You may want to change the user password for security reasons, see [Table 10-1](#) for all parameters.

Table 10-1: Local User and Administrator Password Settings

| Central Provisioning Server | template > parameter |
|--|---|
| Set the minimum length for the administrator password..... | site.cfg > sec.pwd.length.admin |
| Set the minimum length for the user password | site.cfg > sec.pwd.length.user |
| Set the phone's local administrator password | device.cfg > device.auth.localAdminPassword |
| Set the phone's local user password | device.cfg > device.auth.localUserPassword |

Web Configuration Utility

To change the user or administrator password, navigate to **Settings > Change Password**. To change the administrator password, you must log in to the Web configuration utility as an administrator.

Local Phone User Interface

To change the administrator password, navigate to **Menu > Settings > Advanced**, enter the current administrator password, and select **Admin Settings > Change Admin Password**.

To change the User Password, navigate to **Menu > Settings > Advanced**, enter the current user or administrator password, and select **Change User Password**.

Incoming Signaling Validation

You can choose from three optional levels of security for validating incoming network signaling:

- Source IP address validation
- Digest authentication
- Source IP address validation and digest authentication

See [Table 10-2: Incoming Signal Validation](#) for the parameters that specify the validation type, method, and the events you want to validate.

Table 10-2: Incoming Signal Validation

| Central Provisioning Server | template > parameter |
|---|--|
| Specify what type of validation to perform | sip-interop.cfg > volp.SIP.requestValidation.x.method |
| Set the name of the method for which validation will be applied
..... | sip-interop.cfg > volp.SIP.requestValidation.x.request |
| Determine which events within the Event header should be validated
..... | sip-interop.cfg > volp.SIP.requestValidation.x.request.y.event |

Configuration File Encryption

You can encrypt configuration files (excluding the master configuration file), contact directories, and configuration override files can all be encrypted.

For more details on encrypting configuration files, see [Encrypting Configuration Files](#).

You can determine whether encrypted files are the same as unencrypted files and use the SDK to facilitate key generation. Use [Table 10-3: Configuration File Encryption](#) to locate the parameters used to encrypt files. For more information about encrypting configuration files, see [Encrypting Configuration Files](#).

Table 10-3: Configuration File Encryption

| Central Provisioning Server | template > parameter |
|---|--|
| Specify if configuration files uploaded from the phone to the provisioning server should be encrypted
..... | site.cfg > sec.encryption.upload.config |
| Specify if the contact directory is encrypted when it is uploaded from the phone to the provisioning server
..... | site.cfg > sec.encryption.upload.dir |
| Specify if the configuration overrides file should be encrypted when it is uploaded from the phone to the server
..... | site.cfg > sec.encryption.upload.overrides |
| Specify an encryption key so the phone can download encrypted files from the provisioning server
..... | device.cfg > device.sec.configEncryption.key |

Digital Certificates

Polycom phones are installed with a Polycom-authenticated RSA certificate. You can use this certificate to create a secure connection between phone and server when initiating Transport Layer Security (TLS) communications over protocols such as HTTPS and SIP. You can download the Polycom Root CA from <http://pki.polycom.com/>. Note that the certificate is set to expire on March 9, 2044.



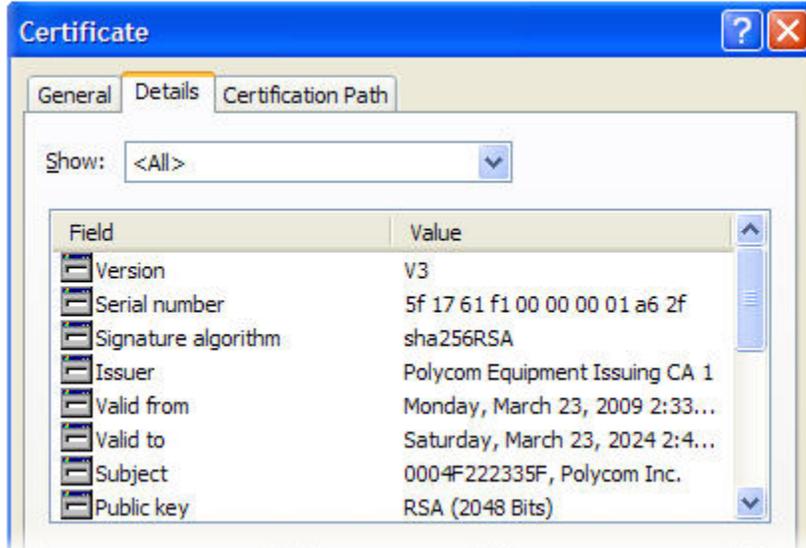
Web Info: Digital Certificates on Polycom Phones

Beginning May 2009, Polycom is installing digital credentials on all SoundPoint IP phone models at the manufacturing facility. For details, see [Technical Bulletin 37148: Device Certificates on Polycom SoundPoint IP, SoundStation IP, and VVX 1500 Phones](#)

Polycom uses the X.509 standard, which defines what information can go into a certificate. An X.509 digital certificate is a digitally signed statement. All X.509 certificates have the following fields, in addition to the signature:

- **Version**—This identifies which version of the X.509 standard applies to this certificate, which in turn affects what information can be specified in the certificate.
- **Serial Number**—The entity that created the certificate is responsible for assigning it a serial number to distinguish it from other certificates it issues.
- **Signature Algorithm Identifier**—This identifies the algorithm used by the Certificate Authority (CA) to sign the certificate.
- **Issuer Name**—The X.500 name of the entity that signed the certificate. This is normally a CA. Using this certificate means trusting the entity that signed this certificate.
- **Validity Period**—Each certificate is valid for a limited amount of time. This period is described by a start date and time and an end date and time, and can be as short as a few seconds or almost as long as a century.
- **Subject Name**—The name of the entity whose public key the certificate identifies. This name uses the X.500 standard, so it is intended to be unique across the Internet.
- **Subject Public Key Information**—This is the public key of the entity being named, together with an algorithm identifier that specifies to which public key cryptographic system this key belongs and any associated key parameters.

The following is an example of a Polycom device certificate when viewed in a browser.



The device certificate and associated private key are stored on the phone in its non-volatile memory as part of the manufacturing process. For more information on digital certificates, see [Public Key Infrastructure \(X.509\)](#) and [RFC 2459: Internet X.509 Public Key Infrastructure](#).



Web Info: Using Custom Certificates With Polycom Phones

As of UC Software 4.0.0, you can install custom device certificates on your Polycom phones. These certificates are installed in the same way custom CA certificates are installed. See [Technical Bulletin 17877: Using Custom Certificates With Polycom Phones](#).

To determine if there is a device certificate on a Polycom phone:

- 1 Press the **Menu** key and select **Settings > Advanced > Admin Settings > TLS Security > Custom Device Certificates**.

You can view the Polycom device certificate on the phone at **Menu > Status > Platform > Phone**.

- 2 Press the **Info** soft key to view the certificate.

One of the following messages will be displayed:

- **Device Certificate: Installed** or **Device Certificate: Factory Installed** is displayed if the certificate is available in flash memory, all the certificate fields are valid (listed above), and the certificate has not expired.
- **Device Certificate: Not Installed** is displayed if the certificate is not available in flash memory (or the flash memory location where the device certificate is to be stored is blank).

- **Device Certificate: Invalid** is displayed if the certificate is not valid.



Note: Device Certificate Shown as Self-Signed

Some Polycom phones manufactured after December, 2011 report the device certificate as 'self-signed' and not as 'Factory Installed'. The difference indicates that different issuing CAs were used to generate the certificates. As long as the authenticating server trusts the Polycom Root CA that issued these certificates, the phones will operate correctly.

Generating a Certificate Signing Request

You may need a certificate to perform a number of tasks, for example, multiple TLS authentication. To obtain a certificate you need to:

- Request a certificate from a Certificate Authority (CA) by creating a certificate signing request (CSR).
- Forward the CSR to a CA to create a certificate. If your organization doesn't have its own CA, you will need to forward the CSR to a company like Symantec. If successful, the CA will send back a certificate that has been digitally signed with their private key.

After you receive the certificate, you can download it to the phone:

- Using a configuration file
- Through the phone's user interface
- Through the Web Configurable Utility

To generate a certificate signing request on a Polycom phone:

- 1 Navigate to **Settings > Advanced > Admin Settings > Generate CSR**.

When prompted, enter the administrative password and press the **Enter** soft key. The default administrative password is **456**.

From the Generate CSR Screen, enter information as shown next. You must fill in the Common Name field - the Organization, Email Address, Country, and State fields are optional.

The following figure shows the Generate CSR screen on a VVX 500 phone.



3 Press **Generate**.

A message *CSR generation completed* displays on the phone's screen.

TLS Profiles

The Transport Layer Security (TLS) profiles describe a collection of custom CA and device certificates installed on the Polycom phones and the features where these certificates are used for authentication.

Your phone can trust certificates issued by widely recognized certificate authorities when trying to establish a connection to a provisioning server for application provisioning. There are a number of parameters you can use to configure TLS Profiles listed in [Table 10-4: Configuring TLS Platform Profiles and TLS Application Profiles](#). For the complete list of trusted Certificate Authorities, see [Trusted Certificate Authority List](#).

Custom CA and device certificates can be added to the phone and set up to be used by different features. For example, the phone's factory-installed or custom device certificate could be used for authentication when phone provisioning is performed by an HTTPS server. A custom CA certificate could also be used when accessing content through the microbrowser or browser.

Once you install certificates on the phone, you can determine which TLS Platform Profiles or TLS Application Profiles will 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.

Lastly you must choose which TLS platform profile or application profile will be used for each TLS Application. The profiles can be used for phone provisioning, with the applications running on the microbrowser and browser, and for 802.1X, LDAP, and SIP authentication. Some applications, such as Syslog, can only use a TLS Platform Profile, not a TLS Application Profile. See [<TLS/>](#) for the list of applications.

For more information on device (or digital) certificates installed on the phones at the factory, see [Digital Certificates](#).



Web Info: Using Custom Certificates

For more information on using custom certificates, see [Technical Bulletin 17877: Using Custom Certificates With Polycom Phones](#).

The following table 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 `.caCertList1`.

Table 10-4: Configuring TLS Platform Profiles and TLS Application Profiles

| Central Provisioning Server | template > parameter |
|---|---|
| TLS Platform Profile Parameters (use 2 at the end of each parameter (instead of 1) to set up platform profile 2) | |
| Specify which CA certificates to use | device.cfg > device.sec.TLS.profile.caCertList1 |
| Specify the cipher suite | device.cfg > device.sec.TLS.profile.cipherSuite1 |
| Select the default cipher suite or a custom cipher suite | device.cfg
> device.sec.TLS.profile.cipherSuiteDefault1 |
| Specify a custom certificate | device.cfg > device.sec.TLS.customCaCert1 |
| Specify which device certificates to use..... | device.cfg > device.sec.TLS.profile.deviceCert1 |
| TLS Application Profile Parameters | |
| Specify which CA certificates to use | site.cfg > sec.TLS.profile.x.caCert.* |
| Specify the cipher suite | site.cfg > sec.TLS.profile.x.cipherSuite |
| Select the default cipher suite or a custom cipher suite ... | site.cfg > sec.TLS.profile.x.cipherSuiteDefault |
| Specify a custom certificate | site.cfg > sec.TLS.customCaCert.x |
| Specify which device certificates to use..... | site.cfg > sec.TLS.profile.x.deviceCert |
| Specify the custom device key..... | site.cfg > sec.TLS.customDeviceKey.x |

Web Configuration Utility

To install CA or device certificates and configure TLS profiles, navigate to **Settings > Network > TLS** and expand the **Certificate Configuration** and **TLS Profiles** menus.

Local Phone User Interface

To install a CA or device certificate, navigate to **Menu > Settings > Advanced > Admin Settings > TLS Security** and select **Custom CA Certificates** or **Custom Device Credentials** and enter the URL of a custom certificate or PEM-encoded certificate.

Once you have configured the certificates, configure a TLS profile. To configure TLS profiles, navigate to **Menu > Settings > Advanced > Admin Settings > TLS Security > Configure TLS Profiles**. Select the profile that you would like to configure, and configure the cipher suite, choose which CA certificates to use, and choose which device certificates to use. The menu options are: **Configure Cipher Suite**, **CA Certificates**, and **Device Certificates**.

This section provides detailed information on:

- [Downloading Certificates to a Polycom Phone](#)
- [Configuring TLS Profiles](#)

Downloading Certificates to a Polycom Phone

You can download certificates to a Polycom phone by specifying a URL where the certificate is currently stored. You can install up to eight CA certificates and eight device certificates on the phone. You can refresh certificates when they expire or are revoked. You can delete any CA certificate or device certificate that you install.



Note: Maximum Size for Certificates

For SoundPoint IP, SoundStation IP, and VVX phones, the maximum certificate size on Platform CA1 is 1536KB and 4KB for Platform CA2.

For SpectraLink 8400 Series wireless handsets, the maximum certificate size on both Platform CA1 and Platform CA2 is 4KB.

To download a certificate to a Polycom phone:

- 1 Navigate to **Menu > Settings > Advanced > Administrative Settings > TLS Security** and select **Custom CA Certificates** or **Custom Device Certificates**.

When prompted, enter the administrative password and press the **Enter** soft key. The default administrative password is **456**.

- 2 Select the **Install** soft key.
- 3 Enter the URL where the certificate is stored.

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

4 Select the **Enter** soft key.

The certificate is downloaded. The certificate's MD5 fingerprint displays to verify that the correct certificate is to be installed.

5 Select the **Accept** soft key.

The certificate is installed successfully.

The appropriate certificate menu displays the certificate's common name.

Configuring TLS Profiles

By default, all Polycom-installed profiles are associated with the default cipher suite and use trusted and widely recognized CA certificates for authentication. You can change the cipher suite, CA certificates, and device certificates for the two platform profiles and the six application profiles. You can then map profiles directly to the features that use certificates.

Table 10-5: Setting a TLS Profile for each TLS Application

| Central Provisioning Server | template > parameter |
|--|--|
| Specify the TLS profile to use for each application (802.1X and Provisioning) | device.cfg > device.sec.TLS.profileSelection.* |
| Specify the TLS profile to use for each application (other applications) | device.cfg > sec.TLS.profileSelection.* |
| Web Configuration Utility | |
| To specify the TLS profile to use for a specific application, navigate to Settings > Network > TLS , and expand the TLS Applications menu. | |
| Local Phone User Interface | |
| To specify the TLS profile to use for a specific application, navigate to Menu > Settings > Advanced > Admin Settings > TLS Security > TLS Applications , select the TLS application, and choose a TLS Profile to use. | |

Supporting Mutual TLS Authentication

Mutual Transport Layer Security (TLS) authentication is a process in which both entities in a communications link authenticate each other. In a network environment, the phone authenticates the server and vice-versa. In this way, phone users can be assured that they are doing business exclusively with legitimate entities and servers can be certain that all would-be users are attempting to gain access for legitimate purposes.

This feature requires that the phone being used has a Polycom factory-installed device certificate or a custom device certificate installed on it. See the section, [Digital Certificates](#).

Prior to SIP 3.2, and in cases where the phones do not have device certificates, the phone will authenticate to the server as part of the TLS authentication, but the server cannot cryptographically authenticate the phone. This is sometimes referred to as Server Authentication or single-sided Authentication.

Mutual TLS authentication is optional and is initiated by the server. When the phone acts as a TLS client and the server is configured to require mutual TLS, the server will request and then validate the client certificate during the handshake. If the server is configured to require mutual TLS, a device certificate and an associated private key must be loaded on the phone.

The device certificate, stored on the phone, is used by:

- HTTPS device configuration, if the server is configured for Mutual Authentication
- SIP signaling, when the selected transport protocol is TLS and the server is configured for Mutual Authentication
- Syslog, when the selected transport protocol is TLS and the server is configured for Mutual Authentication
- Corporate Directory, when the selected transport protocol is TLS and the server is configured for Mutual Authentication
- 802.1X Authentication, if the server is configured for Mutual Authentication (optional for EAP-TLS)



Note: You Cannot Modify the Factory-Installed Certificate or Private Key

At this time, the user will not be able to modify or update the digital certificate or the associated private key installed on the phone during manufacturing. They can install a custom device certificate to be used instead of, or in addition to, the factory-installed certificate.

The Polycom Root CA can be downloaded from <http://pki.polycom.com>. The location of the Certificate Revocation List (CRL)—a list of all expired certificates signed by the Polycom Root CA—is part of the Polycom Root CA digital certificate. If Mutual TLS is enabled, the Polycom Root CA or your organization's CA must be downloaded onto the HTTPS server.

The following operating system/Web server combinations have been tested and verified:

- Microsoft Internet Information Services 6.0 on Microsoft Windows Server 2003
- Apache v1.3 on Microsoft Windows XP



Web Info: Provisioning Using Microsoft Internet Information Services

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 \(Technical Bulletin 52609\)](#).

Configurable TLS Cipher Suites

The phone administrator can control which of cipher suites will be offered/accepted during TLS session negotiation. The phone supports the cipher suites listed in [Table 10-6: TLS Cipher Suites](#), as shown next, and you can use [Table 10-7: Configurable TLS Cipher Suites](#) to configure TLS Cipher Suites. The 'Null Cipher' listed in Table 10-6 is a special case option which will not encrypt the signaling traffic, and is useful for troubleshooting purposes.

Table 10-6: TLS Cipher Suites

| <i>Cipher</i> | <i>Cipher Suite</i> |
|---------------|--|
| ADH | ADH-RC4-MD5, ADH-DES-CBC-SHA, ADH-DES-CBC3-SHA, ADH-AES128-SHA, ADH-AES256-SHA |
| AES128 | AES128-SHA |
| AES256 | AES256-SHA |
| DES | DES-CBC-SHA, DES-CBC3-SHA |
| DHE | DHE-DSS-AES128-SHA, DHE-DSS-AES256-SHA, DHE-RSA-AES128-SHA, DHE-RSA-AES256-SHA |
| EXP | EXP-RC4-MD5, EXP-DES-CBC-SH, EXP-EDH-DSS-DES-CBC-SHA, EXP-DES-CBC-SHA, EXP-ADH-RC4-MD5, EXP-ADH-DES-CBC-SHA, EXP-EDH-RSA-DES-CBC-SHA |
| EDH | EDH-RSA-DES-CBC-SHA, EDH-DSS-DES-CBC3-SHA, EDH-DSS-CBC-SHA |
| NULL | NULL-MD5, NULL-SHA |
| RC4 | RC4-MD5, RC4-SHA |



Tip: Changes to the Default TLS Cipher Suites in UC Software 4.0.0

Changes have been made to the default TLS cipher suites in UC Software 4.0.0. If you created customized TLS cipher suites in a previous release of the UC Software, your changes will be lost unless you backup the configuration files.

Table 10-7: Configurable TLS Cipher Suites

| Central Provisioning Server | template > parameter |
|---|---|
| Specify the global cipher list | site.cfg > sec.TLS.cipherList |
| Specify the cipher list for a specific TLS Platform Profile or TLS Application Profile | site.cfg > sec.TLS.<application>.cipherList |
| Web Configuration Utility | |
| To specify the cipher list for a specific TLS Platform Profile or TLS Application Profile , navigate to Settings > Network > TLS and expand the TLS Profiles menu. | |
| Local Phone User Interface | |
| To specify the cipher list for a specific TLS Platform Profile or TLS Application Profile, navigate to Menu > Settings > Advanced > Admin Settings > TLS Profiles > Configure TLS Profiles , select a profile, and choose Configure Cipher Suite . | |

Secure Real-Time Transport Protocol

Secure Real-Time Transport Protocol (SRTP) provides a way of encrypting audio stream(s) to avoid interception and eavesdropping on phone calls. As described in RFC 3711, both RTP and RTCP signaling may be encrypted using an AES (advanced encryption standard) algorithm. The parameters used to configure SRTP are shown in [Table 10-8: Secure Real Time Transport Protocol](#). When this feature is enabled, phones will negotiate with the other end-point the type of encryption and authentication to use for the session. This negotiation process is compliant with RFC4568 —Session Description Protocol (SDP) Security Descriptions for Media Streams.



Web Info: SRTP RFC Resources

For more information on SRTP, see [RFC 3711](#). For the procedure describing how two phones set up SRTP for a call, see [RFC 4568](#).

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 receiver knows the key to restore the data.

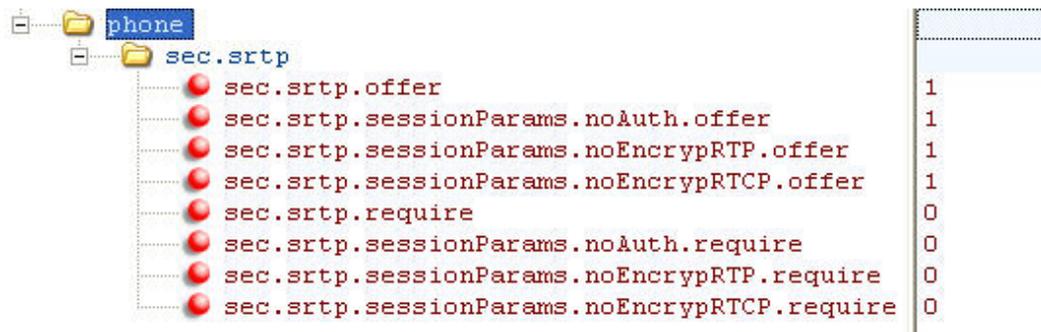
A number of session parameters have been added to enable you to turn off authentication and encryption for RTP and RTCP streams. This is done mainly to reduce the phone's processor usage.

If the call is completely secure (RTP authentication and encryption and RTCP authentication and RTCP encryption are enabled), then the user sees a padlock symbol  appearing in the last frame of the connected context animation (two arrows moving towards each other)

Table 10-8: Secure Real Time Transport Protocol

| Central Provisioning Server | template > parameter |
|--|---|
| Enable SRTP | sip-interop.cfg > sec.srtp.enable |
| Include secure media in SDP of SIP INVITE | sip-interop.cfg > sec.srtp.offer |
| Include crypto in offered SDP | sip-interop.cfg > sec.srtp.offer.* |
| Secure media stream required in all SIP INVITEs | sip-interop.cfg > sec.srtp.require |
| Check tag in crypto parameter in SDP | sip-interop.cfg > sec.srtp.requireMatchingTag |
| Specify if the phone offers and/or requires: RTP encryption, RTP authentication, and RTCP encryption | sip-interop.cfg > sec.srtp.sessionParams.* |

In Example 1, the **srtp_1.cfg** configuration file is shown below:



This would result in an offer (SIP INVITE with SDP) with 8 crypto attributes with the following session parameters:

```
<no session parameters> UNENCRYPTED_SRTCP UNENCRYPTED_S RTP
UNAUTHENTICATED_S RTP
UNAUTHENTICATED_S RTP , UNENCRYPTED_S RTCP UNENCRYPTED_S RTP , UNENCRYPTED_S RTCP
UNAUTHENTICATED_S RTP , UNENCRYPTED_S RTP
UNAUTHENTICATED_S RTP , UNENCRYPTED_S RTP , UNENCRYPTED_S RTCP
```

In the above example, the crypto attributes are ordered “most secure” to “least secure” (more security turned off). The phone receiving this call should chose the most secure crypto it can support based on the SRTP *require* settings in **sip.cfg** and reply with it in the SDP of a 200 OK SIP message.

In Example 2, the **srtp_2.cfg** configuration file is shown below:

| | |
|---|---|
| phone | |
| sec.srtp | |
| sec.srtp.offer | 1 |
| sec.srtp.sessionParams.noAuth.offer | 1 |
| sec.srtp.sessionParams.noEncrypRTP.offer | 1 |
| sec.srtp.sessionParams.noEncrypRTCP.offer | 1 |
| sec.srtp.require | 1 |
| sec.srtp.sessionParams.noAuth.require | 0 |
| sec.srtp.sessionParams.noEncrypRTP.require | 1 |
| sec.srtp.sessionParams.noEncrypRTCP.require | 0 |

This would result in an offer (SIP INVITE with SDP) with 4 crypto attributes with the following session parameters:

```
UNENCRYPTED_SRTP UNENCRYPTED_SRTP , UNENCRYPTED_SRTCP
UNAUTHENTICATED_SRTP , UNENCRYPTED_SRTP
UNAUTHENTICATED_SRTP , UNENCRYPTED_SRTP , UNENCRYPTED_SRTCP
```

In the above example, every crypto includes the UNENCRYPTED_SRTP session parameter because it is required.

If nothing compatible is offered based on the receiving phone’s STRP “require” settings, then the call is rejected or dropped.

Locking the Phone

As of Polycom UC Software 3.3.0, users can lock their phones, and prevent access to the menu or key presses, by pressing the **Lock** soft key or through the phone menu. On the SpectraLink handsets, users can lock their handset through the menu only.



Note: Displaying the Lock Soft Key On Your Phone

You need to enable the enhanced feature key (EFK) feature if you want your phone to display a **Lock** soft key. See [feature.enhancedFeatureKeys.enabled](#).

The following configuration file snippet shows how to display the **Lock** soft key.

| | |
|---------------------------|--------------|
| #comment | |
| phoneLock | |
| phoneLock.enabled | 1 |
| phoneLock.powerUpUnlocked | 1 |
| phoneLock.dndWhenLocked | 1 |
| softkey | |
| softkey.1.enable | 1 |
| softkey.1.label | Lock |
| softkey.1.action | FLockPhone\$ |
| softkey.1.use.idle | 1 |

Once the phone is locked, all user features and access to menus are disabled. The messages **The phone is locked.** and **Authorized calls only.** display on the screen. Incoming calls to the phone may receive a Do Not Disturb message. You can specify the authorized numbers to which users can place calls.

Using the **New Call** soft key, users can place calls using up to five authorized numbers including the emergency number. If the user places a call —using the keypad— to a number that matches an authorized number, the call will proceed. This is to ensure that certain numbers such as emergency numbers can be placed from the phone.

To unlock the phone, the user presses the **Unlock** soft key and enters their password; if it is entered correctly, the phone returns to its normal idle state.

In case the user forgets their password, the system administrator can unlock their 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. See [Table 10-9: Phone Lock](#) for the parameters that configure the phone lock feature.



Note: Shared Lines on Locked Phones

If a locked phone has a registered shared line, calls to the shared line will be displayed on the locked phone and the phone's user will be able to answer the call.

Table 10-9: Phone Lock

| Central Provisioning Server | template > parameter |
|---|--|
| Enable enhanced feature keys | features.cfg > feature.enhancedFeatureKeys.enabled |
| Enable or disable phone lock | features.cfg > phoneLock.enabled |
| Specify an authorized contact (description and value) who can be called while the phone is locked | features.cfg > phoneLock.authorized.* |
| Specify the scenarios when phone lock should be enabled | features.cfg > phoneLock.* |
| <hr/> | |
| Web Configuration Utility | |
| To enable and configure phone lock, navigate to Settings > Phone Lock . | |
| <hr/> | |
| Local Phone User Interface | |
| To lock the phone, press the Lock soft key (if available) or navigate to Settings > Basic > Preferences > Lock Phone . To unlock the phone, press the Unlock soft key and enter the user or administrator password. | |

Locking the Keypad on Your SpectraLink Handset

You can configure your SpectraLink handsets to support a keypad lock feature, as shown in [Table 10-10: Keypad Lock](#). This feature prevents the user from accidentally placing calls. Key presses are ignored until the user unlocks their handset.

Table 10-10: Keypad Lock

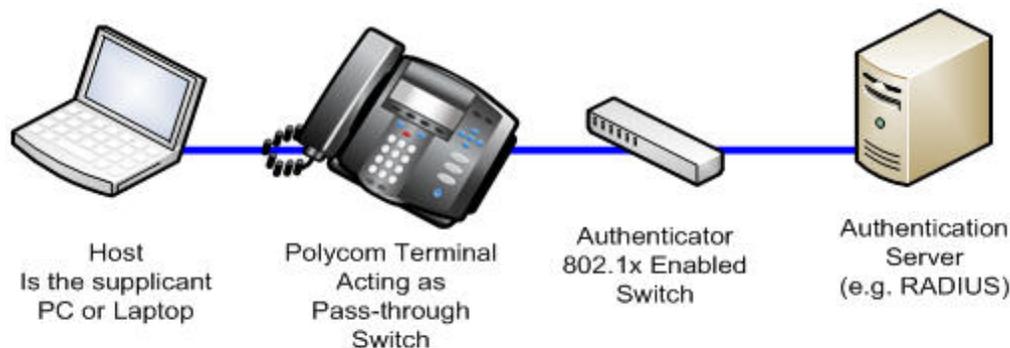
| | |
|--|---|
| Central Provisioning Server | template > parameter |
| Enable or disable keypad lock | features.cfg > keypadLock.enabled |
| Specify how long the phone can be idle before the keypad locks | features.cfg > keypadLock.idleTimeout |
| Web Configuration Utility | |
| To enable or disable keypad lock and set the maximum timeout, navigate to Preferences > Additional Preferences and expand the Keypad Lock menu. | |

Secondary Port Link Status Report

SoundPoint IP phones equipped with a secondary (PC) port can act as a pass-through switch for externally attached devices (such as the Host in [Figure 10-1](#)). The phone informs the network switch (authenticator) of any secondary (PC) port link status changes.

As of Polycom UC Software 3.3.0, Polycom phones include this feature.

Figure 10-1: A Polycom Terminal Acting as a Pass-Through Switch



If you want to configure this feature, see [Table 10-11: Secondary Port Link Status Report](#) for the parameters you will need to set. The SoundPoint IP phones detect an externally connected host connection/disconnection, informing the authenticator switch to initiate the authentication process or drop an existing authentication. This feature ensures that the port authenticated by the externally attached device will be switched to unauthenticated upon device disconnection so

that other unauthorized devices cannot use it. It will also make sure that the externally attached device can move to another port in the network and start a new authentication process. 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 SoundPoint IP secondary (PC) port.

To reduce the frequency of CDP packets, the phone will not send link up status CDP packets before a certain time period. The phone will immediately send all link-down indication to ensure that the port security will not be compromised. The required elapse time —sleep time—between two CDP UPs dispatching will be configurable (see [sec.hostmovedetect.cdp.sleepTime](#)).

If the externally attached device (the Host) supports 802.1X authentication, then the SoundPoint IP phone can send an EAPOL-Logoff on behalf of the device once it is disconnected from the secondary (PC) port. This will inform the authenticator switch to drop the authentication on the port corresponding with the previously attached device.

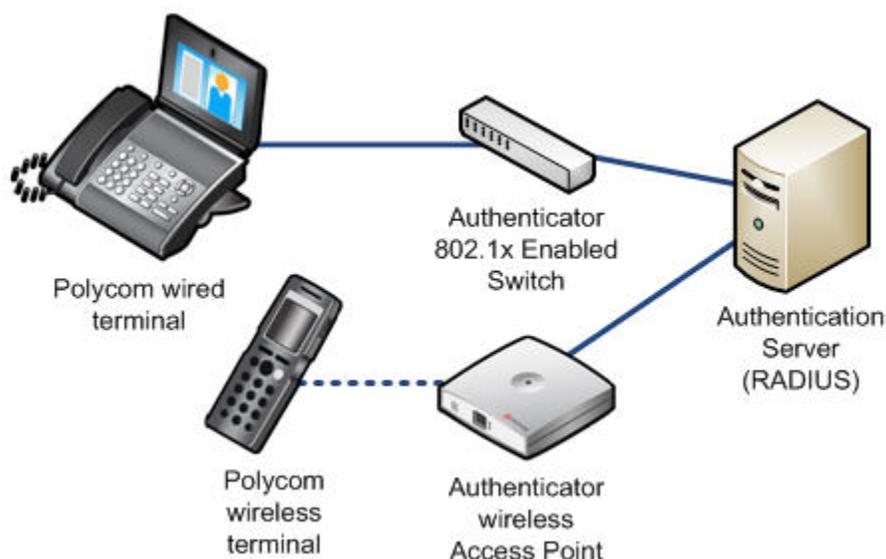
Table 10-11: Secondary Port Link Status Report

| Central Provisioning Server | template > parameter |
|--|---|
| Enable or disable EAPOL logoff | site.cfg > sec.dot1x.eapollogoff.enabled |
| Specify if the LAN port link should be reset or not..... | site.cfg > sec.dot1x.eapollogoff.lanlinkreset |
| Specify the phone should indicate to a host that it has been connected or disconnected to the host's secondary (PC) port | site.cfg > sec.hostmovedetect.cdp.enabled |
| Set the time interval between link-up and link-down reporting | site.cfg
> sec.hostmovedetect.cdp.sleepTime |

Supporting 802.1X Authentication

IEEE 802.1X is a port-based Network Access Control (PNAC). It provides an authentication mechanism to devices trying to attach to a local area network (LAN) or a wireless local area network (WLAN). IEEE 802.1X is based on the Extensible Authentication Protocol (EAP). As of Polycom UC Software 3.3.0, Polycom phones support standard IEEE 802.1X authentication. [Figure 10-2](#) shows a typical 802.1X network configuration with wired and wireless Polycom phones.

Figure 10-2: A Typical 802.1X Network Configuration



Polycom phones support 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



Note: EAP Authentication on SpectraLink Handsets

The SpectraLink handsets support only the EAP-PEAPv0/MSCHAPv2 and EAP-FAST authentication methods.

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 [Table 10-12: Supporting 802.1X Authentication](#) to configure 802.1X Authentication. For more information see [TLS Profiles](#).



Web Info: EAP Authentication Protocol

For more information, see [RFC 3748](#), Extensible Authentication Protocol.

Table 10-12: Supporting 802.1X Authentication

| Central Provisioning Server | template > parameter |
|--|--|
| Enable or disable the 802.1X feature..... | device.cfg > device.net.dot1x.enabled |
| Specify the identity (username) for authentication..... | device.cfg > device.net.dot1x.identity |
| Specify the 802.1X EAP method..... | device.cfg > device.net.dot1x.method |
| Specify the password for authentication | device.cfg > device.net.dot1x.password |
| To enable EAP In-Band Provisioning for EAP-FAST | device.cfg > device.net.dot1x.eapFastInBandProv |
| Specify a PAC file for EAP-FAST (optional) | device.cfg > device.pacfile.data |
| Specify the optional password for the EAP-FAST PAC file | device.cfg > device.pacfile.password |
| <hr/> | |
| Web Configuration Utility | |
| To enable and configure the 802.1X feature, navigate to Settings > Network > Ethernet and expand the Ethernet 802.1X menu. | |
| <hr/> | |
| Local Phone User Interface | |
| To enable 802.1X authentication, navigate to the Ethernet Menu (Menu > Settings > Advanced > Admin Settings > Network Configuration > Ethernet Menu) and select 802.1X Auth . | |
| To configure the 802.1X feature, navigate to the Ethernet Menu and select 802.1X Menu (802.1X Auth must be first set to enabled). | |

Using User Profiles

There are a number of parameters shown in [Table 10-13: User Profiles](#) that enable users to access their personal phone settings from any phone in the organization. This means that users can access their contact directory and speed dials, as well as other phone settings, even if they temporarily change work areas. This feature is particularly useful for remote and mobile workers who do not have a dedicated work space and conduct their business in more than one location. The User Profile feature is also beneficial if an office has a common conference phone. In this case, multiple users could use the phone and access their own settings.

If a user changes any settings while logged in to a phone, the settings will be saved and displayed the next time the user logs in to a phone. When a user logs out, the user's personal phone settings are no longer displayed.

If you set up the User Profile feature, a user can log in to a phone by entering their user ID and password. The default password is **123**.


Tip: Calling Authorized Numbers while Logged Out

You can configure the phones so that anyone can call authorized and emergency numbers when not logged in to a phone. For more information, see [dialplan](#).

If the User Profile feature is set up on your company's phones, users can:

- Log in to a phone to access their personal phone settings.
- Log out of a phone after they finish using it.
- Place a call to an authorized number from a phone that is in the logged out state.
- Change their user password.

When you set up the User Profile feature, you will have to decide whether you want to require users to always log in to a phone. If the User Profile feature is enabled, but not required, users can choose to use the phone as is (that is, without access to their personal settings), or they can log in to display their personal settings. You can specify if a user is logged out of the phone when the phone restarts or reboots, or if they remain logged in.

You can also choose to define default credentials for the phone (see the next section, [Creating a Phone Configuration File](#)). If you specify a default user ID and password, the phone automatically logs 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 is displayed (as defined in the phone's master configuration file on the provisioning server). In this scenario, users will still have the option to log in and view their personal settings.

To set up the User Profile feature, you will need to perform the following procedures on the provisioning server:

- Create a phone configuration file, or update an existing file, to enable the feature's settings.
- Create a user configuration file—called `<user>.cfg`—that specifies the user's password and registration, and other user-specific settings that you want to define.


Tip: Resetting a User's Password

You can reset a user's password by removing the password parameter from the override file. This will cause the phone to use the default password in the `<user>.cfg` file.

After you complete these procedures, update the phone's configuration to affect your changes. The User Profile feature will be ready to use.

Table 10-13: User Profiles

| Central Provisioning Server | template > parameter |
|---|---|
| Enable or disable the user profile feature | site.cfg > prov.login.enabled |
| Specify the amount of time before a non-default user is logged out | site.cfg
> prov.login.automaticLogout |
| Specify the default password for the default user | site.cfg > prov.login.defaultPassword |
| Specify if the phone can have users other than the default user | site.cfg > prov.login.defaultOnly |
| Specify the name of the default user | site.cfg > prov.login.defaultUser |
| Specify the password used to validate the user login | site.cfg > prov.login.localPassword |
| Specify if a user should remain logged in after the handset reboots | site.cfg > prov.login.persistent |
| Specify if a user must log in while the feature is enabled | site.cfg > prov.login.required |

Creating a Phone Configuration File

Create a phone configuration file for the User Login feature, and then add and set the attributes for the feature. Or, if you already have a phone configuration file, update the file to include the User Login parameters you want to change.



Tip: Creating a Default User Password for All Users

Polycom recommends that you create a single default user password for all users.

To define the feature's settings:

- 1 Create a **site.cfg** file for the phone and place it on the provisioning server.
You can base this file on the sample configuration template that is in your software package. To find the file, navigate to **<provisioning server location>/Config/site.cfg**.
- 2 In **site.cfg**, open the `<prov.login/>` attribute, and then add and set values for the user login attributes.

The following example is a sample **site.cfg** file. Your file will contain different values, depending on how you want the feature to work.

| prov. login | |
|------------------------------|-----|
| prov. login. automaticLogout | 0 |
| prov. login. defaultDomain | |
| prov. login. defaultOnly | 0 |
| prov. login. defaultPassword | |
| prov. login. defaultUser | |
| prov. login. enabled | 0 |
| prov. login. localPassword | 123 |
| prov. login. persistent | 0 |
| prov. login. required | 0 |

Creating a User Configuration File

Create a configuration file for each user that you want to be able to log in to the phone. The name of the file will specify the user's login ID. In the file, specify any user-specific settings that you want to define for the user.



Tip: Converting a Phone-Based Deployment to a User-Based Deployment

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**.

To create a user configuration file:

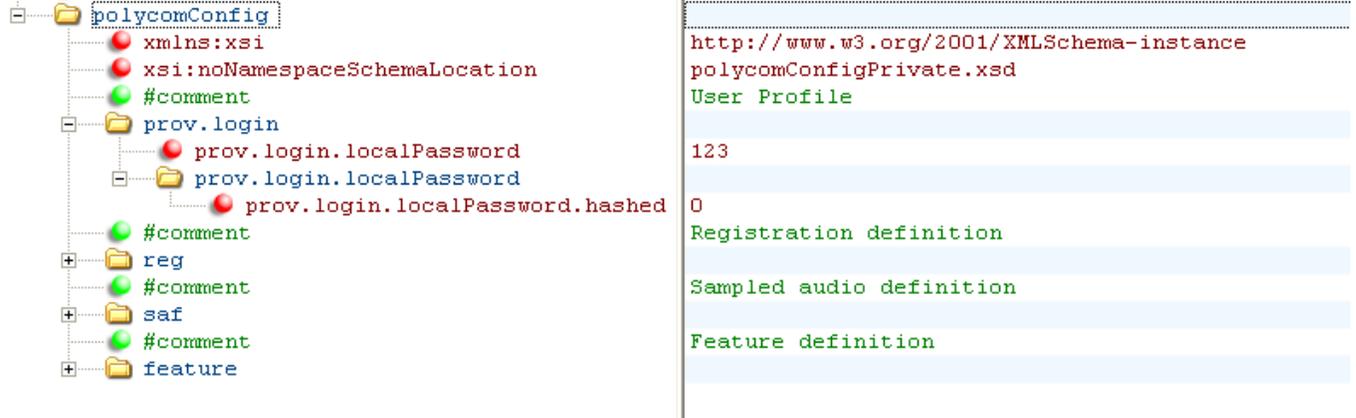
- 1 On the provisioning server, create a user configuration file for each user that will be able to log in to the phone. The name of the file will be the user's ID to log in to the phone. For example, if the user's login ID is **user100**, the name of the user's configuration file is **user100.cfg**.
- 2 In each **<user>.cfg** file, you can add and set values for the user's login password (optional).
- 3 Add and set values for any user-specific parameters, such as:
 - o Registration details (for example, the number of lines the profile will display and line labels).
 - o Feature settings (for example, microbrowser settings).



Caution: Adding User-Specific Parameters

If you add optional user-specific parameters to **<user>.cfg**, add only those parameters that will not cause the phone to restart or reboot when the parameter is updated. For information on which parameters cause the phone to restart or reboot, see [Configuration Parameter Overview](#).

The following is a sample user configuration file.



If a user updates their password or other user-specific settings using the Main Menu on the phone, the updates will be stored in **<user>-phone.cfg**, not **<MACaddress>-phone.cfg**.

If a user updates their Contact Directory while logged in to a phone, the updates will be stored in **<user>-directory.xml**. Directory updates will be displayed each time the user logs in to a phone. For certain phones (for example, the VVX 1500 phone), an up-to-date call lists history will be defined in

<user>-calls.xml. This list will be retained each time the user logs in to their phone.

Configuration parameter precedence (from first to last) for a phone that has the User Profile feature enabled is:

- **<user>-phone.cfg**
- Web Configuration Utility (through a browser)
- Polycom CMA system
- Configuration files listed in the master configuration file (including **<user>.cfg**)
- Default values

Part IV: Troubleshooting and Maintaining your Deployment

Part IV provides you with the information you need to troubleshoot issues with your Polycom® phones and for basic, advanced, audio, video, and user and phone security features.

Part IV consists of the following chapters:

- [Chapter 11: Troubleshooting Your Polycom Phones](#)
- [Chapter 12: Miscellaneous Maintenance Tasks](#)

Chapter 11: Troubleshooting Your Polycom Phones

This chapter shows you some tools and techniques for troubleshooting Polycom® phones running Polycom® UC Software. The phone can provide feedback in the form of on-screen error messages, status indicators, and log files for troubleshooting issues.

This chapter includes information on:

- [Understanding Error Message Types](#)
- [Status Menu](#)
- [Testing Phone Hardware](#)
- [Log Files](#)
- [Managing the Phone's Memory](#)
- [Testing Phone Hardware](#)
- [Uploading a Phone's Configuration](#)
- [Network Diagnostics](#)
- [Ports Used on Polycom Phones](#)

This chapter also addresses phone issues, likely causes, and corrective actions. Issues are grouped as follows:

- [Power and Startup Issues](#)
- [Dial Pad Issues](#)
- [Screen and System Access Issues](#)
- [Calling Issues](#)
- [Display Issues](#)
- [Audio Issues](#)
- [Licensed Feature Issues](#)
- [Upgrading Issues](#)
- [SoundStation Duo Failover Issues](#)

Review the latest *UC Software Release Notes* on the [Polycom UC Software Support Center](#) for known problems and possible workarounds. If a problem is not listed in this chapter or in the latest *Release Notes*, contact your Certified Polycom Reseller for support.

Understanding Error Message Types

Several types of errors can occur while the phone is booting. If an error occurs, the phone will inform you by displaying an error message. Errors can affect how the phone boots up. If the error is fatal, the phone will not be able to boot until the error is resolved. If the error is recoverable, the phone will continue to boot but the phone's configuration may change.

Updater Error Messages

Most of the following errors will be logged to the phone's boot log. However, if you are having trouble connecting to the provisioning server, the phone will likely not be able to upload the boot log.

Failed to get boot parameters via DHCP

The phone does not have an IP address and therefore cannot boot. Check that all cables are connected, the DHCP server is running, and that the phone has not been set to a VLAN that is different from the DHCP server. Check the DHCP configuration.

Application <file name> is not compatible with this phone!

When the Updater displays the error 'The application is not compatible', an application file was downloaded from the provisioning server but cannot be installed on this phone. This issue can usually be resolved by finding a software image that is compatible with the hardware or the BootROM and installing it on the provisioning server. Be aware that there are various different hardware and software dependencies.

Could not contact boot server using existing configuration

The phone could not contact the provisioning server, but the causes may be numerous. It may be a cabling issue, it may be related to DHCP configuration, or it could be a problem with the provisioning server itself. 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!

This message indicates that the phone has no application stored in device settings, that the phone could not download an application, and that the phone cannot boot. To resolve this issue, you must download compatible Polycom UC Software to the phone using one of the supported provisioning protocols. You need to resolve the issue of connecting the phone to the provisioning server and provide a compatible software image on the provisioning server. This error is fatal, but recoverable.

Polycom UC Software Error Messages

The warning notification feature, added in UC Software 4.0.0, provides users a visual indication that one or more error conditions exist. When the warning notification displays, users will see:

- An informative message when the warning is first detected

- An icon in the status bar on the idle display, as shown next
 - On SoundPoint and SoundStation phones 
 - On VVX 500 and 1500 phones 
 - On SpectraLink handsets 
- A persistent list of current warnings, which can be viewed from **Status > Diagnostics > Warnings**

Config file error: Files contain invalid params: <filename1>, <filename2>,...

Config file error: <filename> contains invalid params.

The following contain pre-3.3.0 params: <filename>

These messages display if any of the following pre-Polycom UC Software 3.3.0 parameters are found in the configuration files:

- tone.chord.ringer.x.freq.x
- se.pat.callProg.x.name
- ind.anim.IP_500.x.frame.x.duration
- ind.pattern.x.step.x.state
- feature.2.name
- feature.9.name

This message also appears if any configuration file contains:

- More than 100 unknown parameters, or
- More than 100 out-of-range values, or
- More than 100 invalid values.

Update the configuration files to use the correct parameters, see [Configuration Parameter Overview](#) for details.

CMA Presence not registered.

CMA Directory not registered.

CMA provisioning error.

CMA authentication failed.

These messages may display if a VVX phone is having connection issues with the Polycom Converged Management Application™ (CMA™) system that provisioned it. For more information about provisioning using a Polycom CMA system, see the latest [Polycom CMA System Operations Guide](#).

Insufficient Bandwidth

This message displays if a SpectraLink handset has a poor network connection.

Invalid Regulatory Domain

This message will display on SpectraLink 8400 Series handsets if you set the regulatory domain on your handset to an incorrect regulatory domain for your location. If you see this message, press the *Details* soft key to get additional information about the invalid setting and to find out what are valid settings. If an invalid regulatory domain is set, the handset's radio will be disabled. For example, the valid regulatory domain for the US is 01; if the regulatory domain is set to 10 (New Zealand), then this error is generated and the radio is disabled.

Invalid Regulatory Domain Setting

This message will display on SpectraLink 8400 Series handsets if some of your handset settings are deemed incorrect according to the regulatory domain for your location. Each domain has its own set of restrictions such as TX power limits and sub-bands. If one of these settings is not within the restrictions, an error message displays with the details about which setting is incorrect. If an invalid regulatory domain setting is detected, the handset's radio is not disabled, but the restriction is enforced. For example, this error will be generated if the regulatory domain is set to 01 and TX power is set to P7 for one of the sub-bands.

Line: Unregistered

This message displays if a line fails to register with the call server.

Login credentials have failed. Please update them if information is incorrect.

This message displays when the user enters incorrect login credentials (**Status > Basic > Login Credentials**).

Missing files, config. reverted

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 will also display if the files listed in the **<MAC Address>.cfg** file are not present on the provisioning server.

Network Authentication Failure

This message displays if 802.1X authentication with the Polycom phone fails. The error codes shown in [Table 11-21-2: Managing the Phone Features](#) will display on the phone's screen—if the **Details** soft key is selected—and in the log files:

Table 11-1: Event Codes and Descriptions

| <i>Event Code</i> | <i>Description</i> | <i>Comments</i> |
|-------------------|--------------------|---|
| 1 | Unknown events | This includes any event listed in this table. |

| <i>Event Code</i> | <i>Description</i> | <i>Comments</i> |
|-------------------|--|--|
| 2 | Mismatch in EAP Method type
Authenticating server's list of EAP methods does not match with clients'. | |
| 30xxx | TLS Certificate failure
The TLS certificate-related failures. "xxx" when having a non-zero value, is the standard TLS alert message code. For example, if a bad/invalid certificate (on the basis of its signature and/or content) is presented by the phone, "xxx" will be 042. If the exact reason for the certificate being invalid is not known, then the generic certificate error code will be xxx=000. | See section 7.2 of RFC 2246 for further TLS alert codes and error codes. |
| 31xxx | Server Certificate failure
Certificate presented by the server is considered invalid.
"xxx" can take the following values: <ul style="list-style-type: none"> • 009 - Certificate not yet Valid • 010 - Certificate Expired • 011 - Certificate Revocation List (CRL) not yet Valid • 012 - CRL Expired | |
| 4xxx | Other TLS failures
This is due to TLS failure other than certification related errors. The reason code (the TLS alert message code) is represented by "xxx". For example, if the protocol version presented by the server is not supported by the phone, then xxx will be 70, and the EAP error code will be 4070. | See section 7.2 of RFC 2246 for further TLS alert codes and error codes. |

Network link is down

Since the Polycom phones do not have an LED indicating network LINK status like many networking devices, link failures are indicated with the message

'Network link is down'. This message will be shown on the screen whenever the phone is not in the menu system and will persist until the link problem is resolved. Call related functions and the soft keys and line keys are disabled when the network is down; however the menu works.

Wi-Fi: No APs Found

This message displays on SpectraLink handsets if the handset is unable to connect find an access point (AP) on the wireless network.

Status Menu

Debugging of a single phone may be possible by examining of the phone's status menu. Press **Menu**, select **Status**, and press the **Select** soft key to view the Status menu. Scroll to one of the Status menu items and press the **Select** soft key. Each of the menu items is explained next.



Troubleshooting: I Can't Find the Status Menu on my SpectraLink Handset

To view the **Status** menu on a SpectraLink handset, navigate to **Menu > Settings > Status**.

Under the **Platform** menu, you can get details on the phone's serial number or MAC address, the current IP address, the Updater version, the application version, the name of the configuration files in use, and the address of the provisioning server.

In the **Network** menu, you can find information about the TCP/IP Setting, Ethernet port speed, connectivity status of the PC port (if it exists), and statistics on packets sent and received since last boot. You can also find out the last time the phone rebooted. The Call Statistics screen shows packets sent and received on the last call.

The **Lines** menu will show you details about the status of each line that has been configured on the phone.

Finally, the **Diagnostics** menu offers a series of hardware tests to verify correct operation of the microphone, speaker, handset, and third party headset, if present. You can also test that each of the keys on the phone is working, and display the function assigned to each of the keys in the configuration. You will also find useful information on any access points (APs) that SpectraLink handsets are connected to. This is also where you can test the LCD for faulty pixels.

In addition to the hardware tests, the Diagnostics menu has a series of real-time graphs for CPU, network, and memory use that can be helpful for diagnosing performance issues.

Log Files

Polycom phones will log various events to files stored in the flash file system and will periodically upload these log files to the provisioning server. The files are stored in the phone's home directory or a user-configurable directory. You can also configure a phone to send log messages to a syslog server. If a phone was provisioned by a Polycom CMA system, log messages will be sent to the Polycom CMA system.

There is one log file for the Updater and one for the UC Software. When a phone uploads its log files, they are saved on the provisioning server with the MAC address of the phone prepended to the file name. For example, **0004f200360b-boot.log** and **0004f200360b-app.log** are the files associated with MAC address 00f4f200360b. The Updater (boot) log file is uploaded to the provisioning server after every reboot. The application log file is uploaded periodically or when the local copy reaches a predetermined size. If the Updater was updated (and the file system is cleared) on SoundStation IP 6000 phones, the phone's current **app.log** is uploaded to the provisioning server as **MAC-appFlash.log**. For more information on log file contents, see [lineKey](#).

Both log files can be uploaded on demand using a multiple key combination described in [Multiple Key Combinations](#). The phone uploads four files, namely, **mac-boot.log**, **app-boot.log**, **mac-now-boot.log**, and **mac-now-app.log**. The *-now-*logs are uploaded manually unless they are empty.

The amount of logging that the phone performs can be tuned for the application to provide more or less detail on specific components of the phone's software. For example, if you are troubleshooting a SIP signaling issue, you are not likely interested in DSP events. Logging levels are adjusted in the configuration files or via the Web Configuration Utility. You should not modify the default logging levels unless directed to by Polycom Customer Support. Inappropriate logging levels can cause performance issues on the phone.

In addition to logging events, the phone can be configured to automatically execute command-line instructions at specified intervals that output run-time information such as memory utilization, task status, or network buffer contents to the log file. These techniques should only be used in consultation with Polycom Customer Support.

Logging Options

Each of the components of the Polycom UC Software is capable of logging events of different severity. This allows you to capture lower severity events in one part of the application, and high severity events for other components.

The parameters for log level settings are found in the **techsupport.cfg** configuration file. They are `log.level.change.module_name`. Log levels range from 1 to 6 (1 for the most detailed logging, 6 for critical errors only). There are many different log types that can be adjusted to assist with the investigation of different problems. The exact number of log types is dependent on the phone model.

When testing is complete, remember to remove the configuration parameter from the configuration files.

There are other logging parameters, describe next, that you may wish to modify. Changing these parameters will not have the same impact as changing the logging levels, but you should still understand how your changes will affect the phone and the network.

- `log.render.level`—Sets the lowest level that can be logged (default=1)
- `log.render.file.size`—Maximum size before log file is uploaded (default=32 kb)

- `log.render.file.upload.period`—Frequency of log uploads (default is 172800 seconds = 48 hours)
- `log.render.file.upload.append`—Controls whether log files on the provisioning server are overwritten or appended, not supported by all servers (default=1 so files are appended)
- `log.render.file.upload.append.sizeLimit`—Controls the maximum size of log files on the provisioning server (default=512 kb)
- `log.render.file.upload.append.limitMode`—Control whether to stop or delete logging when the server log reaches its maximum size (default=delete)

Scheduled Logging

Schedules logging is a powerful tool that can help you troubleshoot issues that occur after the phone has been operating for some time.

The output of these instructions is written to the application log, and can be examined later (for trend data).

The parameters for scheduled logging are found in the **techsupport.cfg** configuration file. They are `log.sched.module_name`.

For an example of a configuration file and the resulting log file, see [Figure 11-1: Scheduled Logging Log File](#), shown next.

Figure 11-1: Scheduled Logging Log File



```

0522163019|slog |4|01|#####
0522163033|slog |4|01|#####
0522163033|slog |4|01|Running showCpuLoad
0522163033|slog |4|01|Cpu load is 6.0%, and the average is 57.6%.
0522163033|slog |4|01|#####
0522163033|slog |4|01|#####
0522163033|slog |4|01|Running memShow
0522163033|slog |4|01| status      bytes      blocks      avg block    max block
0522163033|slog |4|01| -----
0522163033|slog |4|01|current
0522163033|slog |4|01| free      9410608      65          144778      9257824
0522163033|slog |4|01| alloc     11147888     31569       353         -
0522163033|slog |4|01|cumulative
0522163033|slog |4|01| alloc     18961376     58186       325         -
0522163033|slog |4|01|#####
0522163048|slog |4|01|#####
0522163048|slog |4|01|Running showCpuLoad
0522163048|slog |4|01|Cpu load is 6.0%, and the average is 47.1%.
0522163048|slog |4|01|#####
0522163048|slog |4|01|#####
0522163048|slog |4|01|Running memShow
0522163048|slog |4|01| status      bytes      blocks      avg block    max block
0522163048|slog |4|01| -----
0522163048|slog |4|01|current

```

Manual Log Upload

If you want to look at the log files without having to wait for the phone to upload them (which could take as long as 24 hours or more), initiate an upload by pressing the correct multiple key combination on the phone (see Multiple Key Combinations).

When the log files are manually uploaded, the word *now* is inserted into the name of the file, for example, **0004f200360b-now-boot.log**.

Reading a Boot Log File

See [Figure 11-2: Boot Log](#) for an example of a boot log file:

Figure 11-2: Boot Log

```

0100000000|so |4|00|+++ Note that bootrom log times are in GMT +++
0100000000|cfg |4|00|Initial log entry
0100000000|copy |3|00|Initial log entry
0100000000|hw |4|00|Initial log entry.
0100000000|ethf |4|00|Initial log entry.
0522182911|wdog |4|00|Initial log entry
0522182911|cdp |3|00|CDP is DISABLED.
0522182911|so |3|00|Platform: Model=SoundPoint IP 450, Assembly=2345-12450-001 Rev=3
0522182911|so |3|00|Platform: Board=2345-12450-001 2
0522182911|so |3|00|Platform: MAC=0004f21db094, IP=Resolving, Subnet Mask=Resolving
0522182911|so |3|00|Platform: BootBlock=2.8.1 (12450_001) 04-Jun-08 17:04
0522182911|so |3|00|Application, main: Label=BOOT, Version=4.1.2.0009 20-Jul-08 21:57
0522182911|so |3|00|Application, main: P/N=3150-11069-412
0522182911|app1 |4|00|Initial log entry.
0522182912|so |3|00|Link status is Net up Speed 100 full Duplex, PC down.
0522182916|cdp |3|00|CDP received a response from a switch. CDP enabled.
0522182916|cdp |3|00|Native VLAN Id is 1
0522182916|cdp |3|00|No Auxiliary VLAN found
    
```

The following Figure 11-3: Boot Failure Messages shows a number of boot failure messages:

Figure 11-3: Boot Failure Messages

```

0522183251|app1 |3|00|DNS domain is Vancouver.polycom.com.
0522183251|cfg |3|00|Beginning to provision phone
0522183251|copy |3|00|'ftp://plcmspip:***@172.23.2.92/2345-12450-001.bootrom.ld' from
0522183251|copy |4|00|Download of '2345-12450-001.bootrom.ld' FAILED on attempt 1 (addr
0522183251|copy |4|00|Server '172.23.2.92' said '2345-12450-001.bootrom.ld' is not pres
0522183251|cfg |4|00|Could not get all 512 bytes of the header
0522183251|copy |3|00|'ftp://plcmspip:***@172.23.2.92/bootrom.ld' from '172.23.2.92'
0522183251|copy |4|00|Download of 'bootrom.ld' FAILED on attempt 1 (addr 1 of 1)
0522183251|copy |4|00|Server '172.23.2.92' said 'bootrom.ld' is not present
0522183251|cfg |4|00|Could not get all 512 bytes of the header
0522183251|cfg |3|00|bootROM file not present on boot server
0522183251|copy |3|00|'ftp://plcmspip:***@172.23.2.92/0004f21db094.cfg' from '172.23.2
0522183251|copy |4|00|Download of '0004f21db094.cfg' FAILED on attempt 1 (addr 1 of 1)
0522183251|copy |4|00|Server '172.23.2.92' said '0004f21db094.cfg' is not present
0522183251|copy |3|00|Update of '/ffs0/init.mac' failed, leaving local copy intact
0522183251|copy |3|00|'ftp://plcmspip:***@172.23.2.92/0000000000000000.cfg' from '172.23.2
0522183251|copy |3|00|Download of '0000000000000000.cfg' succeeded on attempt 1 (addr 1 of
    
```

Reading an Application Log File

The following [Figure 11-4: Application Log File](#) shows portions of an application log file:

Figure 11-4: Application Log File

```

0522184554|log  |*|01|Initial log entry. Current logging level 4
0522184554|so  |*|01|Initial log entry. Current logging level 3
0522184554|so  |*|01|----- Initial log entry -----
0522184554|so  |*|01|Platform: Model=SoundPoint IP 450, Assembly=2345-12450-001 Rev=
0522184554|so  |*|01|Platform: MAC=0004f21db094, IP=172.23.61.141, Subnet Mask=255.2
0522184554|so  |*|01|Platform: BootBlock=2.8.1 (12450_001) 04-Jun-08 17:04
0522184554|so  |*|01|Platform: Bootrom=4.1.2.0009 20-Jul-08 21:57
0522184554|so  |*|01|Application, main: Label=SIP, Version=3.1.3.0439 26-Apr-09 23:5
0522184554|so  |*|01|Application, main: P/N=3150-11530-313
0522184554|wdog |*|01|Initial log entry. Current logging level 4
0522184554|ethf |*|01|Initial log entry. Current logging level 4
0522184554|so  |5|01|utilCertificateInit failed.
0522184554|hw  |*|01|Initial log entry. Current logging level 4
0522184554|ares |*|01|Initial log entry. Current logging level 4
0522184554|dns  |*|01|Initial log entry. Current logging level 3
0522184554|cfg  |*|01|Initial log entry. Current logging level 3

0522114602|so  |*|01|System Info Reports:
0522114602|so  |*|01| CPU is TNETV1055/C55x, rev 2 running at 150MHz with memory at 3
0522114602|so  |*|01| Board is identified as PolycomSoundPointIP-SPIP_450.
0522114602|so  |*|01| DRAM_LO: 0x94000000. DRAM_SIZE: 32 MB
0522114602|so  |*|01| Clocks are VBUSP: 125MHz, VBUS: 75MHz, USB: 25MHz, LCD: 20MHz,
0522114602|key  |*|01|Initial log entry. Current logging level 4
0522114602|ht  |*|01|Initial log entry. Current logging level 4
0522114602|httpd |*|01|Initial log entry. Current logging level 4
0522114602|ssps |*|01|Application, comp. 1: Label=PolyDSP Titan Mem1 FS5 (G.729), Verso

0522185324|cfg  |3|01|Prm|Check of configuration files succeeded
0522185324|cfg  |3|01|Prm|Phone successfully provisioned
0522185324|cfg  |*|01|Prm|Configuration file "001-phone1.cfg" is from template phone1
0522185324|cfg  |*|01|Prm|Configuration file "001-phone1.cfg" SHA1 digest: B712DCCA39
0522185324|cfg  |*|01|Prm|Configuration file "001-sip.cfg" is from template sip.cfg,
0522185324|cfg  |*|01|Prm|Configuration file "001-sip.cfg" SHA1 digest: B4E4534529797
0522185324|so  |3|01|Success provisioning.

0522120608|ldap |*|01|Initial log entry. Current logging level 4
0522120608|ldap |4|01|ldap: Not Enabled
0522120608|ldap |4|01|cDynamicData::cDynamicData:cDynamicData:Failed
0522120608|efk  |*|01|Initial log entry. Current logging level 4
0522120608|so  |*|01|[SoNcasC]: App-Ctx (6045551234) [0-6045551234]
0522120608|sip  |4|01|NAPTR query for host 'as-test' returned no results
0522120608|app1 |*|01|[InitializeBacklightIntensity] m_nDefaultMin = 0, m_nDefaultLow
0522120608|sip  |4|01|Registration failed User: 6045551234, Error Code:404 Not Found
0522120608|cfg  |4|01|Edit|Error 0x380003 attempting stat of /ffs0/local/0004f21db094-

```



Caution: Passwords Appear in cfg Log File

Passwords will appear in a level 1 cfg log file.

Reading a Syslog File

The following [Figure 11-5: Syslog file](#) shows a portion of a syslog log file. Note that the messages look identical to the normal log except for the addition of a timestamp and IP address:

Figure 11-5: Syslog file

```

Jan  0 00:00:00 172.23.7.249 0100000000|so  |4|00|----- Initial log entry -----
Jan  0 00:00:00 172.23.7.249 0100000000|so  |4|00|+++ Note that bootrom log times are in GMT +++
Jan  0 00:00:00 172.23.7.249 0100000000|cfg  |4|00|Initial log entry
Jan  0 00:00:00 172.23.7.249 0100000000|copy |3|00|Initial log entry
Jan  0 00:00:00 172.23.7.249 0100000000|hw   |4|00|Initial log entry.
Jan  0 00:00:00 172.23.7.249 0100000000|ethf  |4|00|Initial log entry.
Feb 13 01:12:39 172.23.7.249 0213011239|wdog  |4|00|Initial log entry
Feb 13 01:12:39 172.23.7.249 0213011239|cdp   |3|00|CDP is DISABLED.
Feb 13 01:12:39 172.23.7.249 0213011239|so    |3|00|Platform: Model=SoundPoint IP 650, Assembly=2345-126
Feb 13 01:12:39 172.23.7.249 0213011239|so    |3|00|Platform: Board=2345-12600-001 1
Feb 13 01:12:39 172.23.7.249 0213011239|so    |3|00|Platform: MAC=0004f2111511, IP=Resolving, Subnet Mas
Feb 13 01:12:39 172.23.7.249 0213011239|so    |3|00|Platform: BootBlock=2.7.0 (12600_001) 30-May-06 15:5
Feb 13 01:12:39 172.23.7.249 0213011239|so    |3|00|Application, main: Label=BOOT, Version=4.1.0.0219 10
Feb 13 01:12:39 172.23.7.249 0213011239|so    |3|00|Application, main: P/N=3150-11069-410
Feb 13 01:12:39 172.23.7.249 0213011239|appl  |4|00|Initial log entry.
Feb 13 01:12:40 172.23.7.249 0213011240|so    |3|00|Link status is Net down, PC down.
Feb 13 01:12:41 172.23.7.249 0213011241|so    |3|00|Link status is Net up Speed 100 half Duplex, PC down
Feb 13 01:12:41 172.23.7.249 0213011241|cdp   |3|00|CDP is disabled.
Feb 13 01:12:45 172.23.7.249 0213011245|appl  |3|00|DNS resolver servers are '172.23.0.200' '172.23.0.23
Feb 13 01:12:45 172.23.7.249 0213011245|appl  |3|00|DNS resolver search domain is 'vancouver.polycom.com
Feb 13 01:12:45 172.23.7.249 0213011245|appl  |3|00|Bootline: esw(3,0)bootHost:flash e=172.23.7.249:fff
Apr 15 22:32:22 172.23.7.249 0415223222|appl  |3|00|Time has been set from 172.23.0.200 (172.23.0.200).
Apr 15 22:32:22 172.23.7.249 0415223222|appl  |3|00|DHCP returned result 0x3E7 from server 172.23.0.232.
Apr 15 22:32:22 172.23.7.249 0415223222|appl  |3|00|   Phone IP address is 172.23.7.249.
Apr 15 22:32:22 172.23.7.249 0415223222|appl  |3|00|   Subnet mask is 255.255.0.0.
Apr 15 22:32:22 172.23.7.249 0415223222|appl  |3|00|   Gateway address is 172.23.2.240.
Apr 15 22:32:22 172.23.7.249 0415223222|appl  |3|00|   Time server is 172.23.0.200.
Apr 15 22:32:22 172.23.7.249 0415223222|appl  |3|00|   GMT offset is -28800 seconds.

```



Web Info: Using Syslog on Polycom Phones

For more information about syslog, see [Technical Bulletin 17124: Using Syslog on Polycom Phones](#).

Managing the Phone's Memory Resources

Polycom phones are designed to operate optimally in a variety of deployments and real-world environments. Each new software release adds new features and capabilities that require varying degrees of the phone's memory resources. To ensure your phones and their configured features operate smoothly, you will need to check that the phones have adequate available memory resources. If you are using a range of phone features - especially customized or advanced features - you may need to manage phone memory resources. To help you optimize your phone features and memory resources, Polycom provides several tools and troubleshooting tips.

Identifying Symptoms

When the phone memory resources start to run low, you may notice one or more of the following symptoms:

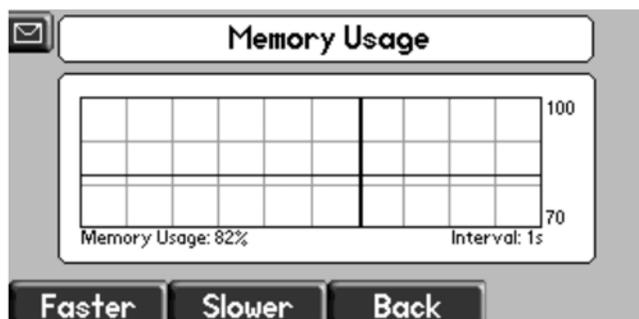
- The phones reboot or freeze up.
- The phones do not download all ringtones, directory entries, backgrounds, or XML dictionary files.
- Applications running in the microbrowser or browser stop or do not run at all.

The next sections show you how to check your phone's available memory and manage the phone features to make phone memory available.

Checking the Phone's Available Memory

You can use two methods to quickly check whether you need to manage your phone's memory. Before you begin checking, load and configure the features and files you want to make available on the phone.

Using the first method, on your phone's keypad or touch pad interface, choose **Status > Diagnostics > Graphs > Memory Usage** as shown next.



Use the *Memory Usage* chart to check what the current Memory Usage amount is. Typically, you want to ensure that the phone is running at less than 95 percent of its available memory.

If the phone is using more than 95 percent of its available memory, you may need to take steps to reduce this amount. For information and tips on freeing memory on the phone, see [Managing the Phone Features](#).

The second method you can use to confirm whether you need to manage your phone's memory is to check the app log files. The app log file is enabled by default and is saved to your provisioning server directory with the MAC address of the phone prepended to the app log file. For example, if the MAC address of your phone is **0004f2203b0**, the app log file name will be **0004f2203b0-app.log**.

Open the app log. If you see the message shown next in [Figure 11-6: Application Log Error Message](#), you may need to manage your phone's memory resources.

Figure 11-6: Application Log Error Message

```

000014.458|dnš   *|00|DNS resolver servers are '172.23.0.200' '172.23.0.239'
000014.458|dnš   *|00|DNS resolver search domain is 'vancouver.polycom.com'
000014.460|cfg   *|00|RT|Primary IP changed to 172.23.70.29 subnet mask 255.255.0.0
000016.412|ib    *|00|Initial log entry. Current logging level 4
000016.414|so    *|00|Network initialized. Starting network tasks.
000016.428|cfg   5|00|Prm|Parameter lcl.ml.lang requested type 2 but is of type 4
000016.428|cfg   5|00|Prm|Type 2 4 0 for parameter lcl.ml.lang is not valid
000016.658|sip   *|00|Fast Boot Measurement Point: Ready for Call, uptime: 16.658 sec.
000016.982|tr69  *|00|Initial log entry. Current logging level 4
000016.984|cfg   *|00|Prov|Starting to update 2345-12670-001.sip.1d
000016.994|app1  *|00|Ctx [1] Registered [true]
000017.004|res   4|00|[ResFinderC]: Minimum free memory reached. 0xaf150.
000017.012|cfg   *|00|Prov|Finished updating configuration.
    
```



Web Info: Reading the App Log Files

For more information on reading the log files see the section [Log Files](#) in *Chapter 11: Troubleshooting your Polycom Phones* of the Polycom® UC Software Administrators' Guide.

Managing the Phone Features

This section provides tips for managing the phone features to conserve phone memory resources. This section is especially useful if you are customizing features or using several advanced features.

If you are using a mixed deployment, such as a combination of SoundPoint IP and SoundStation IP phones, consider configuring each phone model separately with their own features instead of applying all phone features to both phone models. For details on how to use different configurations for each phone model, register to access [White Paper 60806: UC Software Provisioning Best Practices](#), which applies to UCS 3.3.0 or later.

All phone features are designed to operate optimally on Polycom phones. The features listed in [Table 11-2: Managing the Phone Features](#) are all customizable, advanced features that can take up significant memory. Use *Table 11-2* as a reference guide to the amount of memory a feature can use and for tips on balancing features so that you can optimize the phone features you want for your deployment.

Table 11-2: Managing the Phone Features

| Feature | Typical Memory Size |
|--|---|
| Idle Browser | Variable. Optimized to display three or four elements. |
| <p>The idle browser is optimized to display three or four application elements. If you display complex pages that include large table or images, try to display a simplified page. If the page cannot be simplified, try reducing the number of available ringtones or display backgrounds, or disable the main browser.</p> | |

| <i>Feature</i> | <i>Typical Memory Size</i> |
|--|---|
| Custom Idle Display Image | 15KB |
| <p>The average size of Polycom display images is 15KB. If you are using custom images, Polycom recommends limiting the file size to 15KB for images on the idle display. If your phone does not display your custom image and the file size is less than 15KB, try reducing the number of available ringtones or idle display and image backgrounds.</p> | |
| Main Browser | Variable. Optimized to display three or four elements. |
| <p>The main browser is optimized to display three or four application elements. As with the idle browser, try simplifying the content to conserve memory resources. If the content cannot be simplified, try reducing the number of available ringtones or image backgrounds, or disable the idle browser.</p> | |
| Local Contact Directory | 170 bytes per entry |
| <p>Polycom phones are optimized to display four contact attributes to a maximum of 250 contact entries. Each entry averages about 170 bytes of memory. For this reason, Polycom recommends a maximum of 250 contacts on SoundPoint IP 550, 560, and 650 phones.</p> <p>If you need more space for the contact directory, try disabling the idle browser, reducing the number of available ringtones or image backgrounds.</p> | |
| Corporate Directory | Varies by server |
| <p>The Corporate Directory feature is optimized to display five contact attributes up to a maximum of eight on Polycom phones. Because the corporate directory entries are saved to a server, the size of each entry and the corporate directory as a whole will vary with the server you are using. If the phone has difficulty displaying directory search results with more than five attributes, try reducing the number of available ringtones or image backgrounds, or disable the idle browser or main browser.</p> | |
| Ringtones | 16KB |
| <p>Polycom provides a number of audio files for ringtones that are designed to work correctly with the phones. Polycom ringtones can range in size from 30KB to 125KB. If you want to use custom ringtones, Polycom recommends limiting the file size to 16KB. If you want to make more room for custom ringtones, try disabling the idle browser or main browser, or reduce the number of custom or image backgrounds. If you want to make room for other features, try reducing the number of available ringtones.</p> | |
| Background Images | 8 – 32KB |
| <p>Polycom phones are optimized to display background images of about 50KB. If you want to display background images having a file size of more than 50KB or make room for more images, try disabling the idle browser or main Web browser, or reduce the number of available ringtones. If you want to make room for other features, try reducing the number and size of available background images.</p> | |
| Phone Interface Language | 90KB |
| <p>The average size of the <i>SoundPointIPLocalization</i> XML dictionary files for languages that display on the phone's interface is about 90KB. Some of these language files use an expanded character set that can increase the file size to 115KB. To conserve memory resources, Polycom recommends using only those XML language files for the languages you need.</p> | |

| <i>Feature</i> | <i>Typical Memory Size</i> |
|--|----------------------------|
| Web Configuration Utility Interface | 250KB |

The average size of the *languages* XML dictionary files for languages that display on the Web Configuration Utility interface is about 250KB. Some of these language files use an expanded character set that can increase the file size to 370KB. To conserve memory resources, Polycom recommends using only those XML language files for the languages you need.

If you are still having difficulty freeing up sufficient space on your phones, contact [Polycom® Voice Product Support](#).

Testing Phone Hardware

You can view diagnostic information from the **Diagnostics** menu on your phone (**Menu > Status > Diagnostics**).

If you select **Diagnostics > Test Hardware**, you can select one of the following menu items to perform a hardware diagnostic test:

- **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
- **LED Diagnostics** – test the LED lights on your phone
- **Touch Screen Diagnostics** (V VX only) – test the touch screen response

Uploading a Phone's Configuration

As of Polycom UC Software 3.3.0, you can upload the files representing a phone's current configuration. A number of files can be uploaded to the provisioning server, one for every active source as well as the current non-default configuration set.

As of Polycom UC Software 4.0.0, you can upload the phone's configuration through the Web Configuration Utility.

This is primarily a diagnostics tool to help find configuration errors.

To upload the phone's current configuration:

- 1 Navigate to the Upload Configuration menu on the phone (**Menu > Settings > Advanced > Admin Settings > Upload Configuration**).
- 2 Choose to upload the configuration from one of **All Sources**, **Configuration Files**, **Local**, **CMA**, or **Web**.

The CMA option will display on the VVX 1500 phone only. You can select **Device Settings** if you perform this task using the Web Configuration Utility.

3 Press the **Upload** soft key.

The phone uploads the configuration file to the location that you specify in `prov.configUploadPath`. For example, if you select **All Sources**, a file `<MACAddress>update-all.cfg` is uploaded.

Network Diagnostics

In Polycom UC Software 4.0.0, ping and traceroute are added to the phone's diagnostics tools. These diagnostics can be used for troubleshooting network connectivity problems in the wired and wireless worlds.

Both tools are accessible by pressing the **Menu** key and selecting **Status > Diagnostics > Network**.

Enter a URL address (for example, `http://www.google.com`) or any IP address (for example, the system IP address or any other phone's IP address), and then press the **Enter** soft key.

Ports Used on Polycom Phones

See [Table 11-3: Ports used by Polycom Phones](#) for a list of the ports currently used by the Polycom UC Software.

Table 11-3: Ports used by Polycom Phones

| <i>Port Number</i> | <i>Protocol</i> | <i>Outgoing</i> | <i>Incoming</i> | <i>UDP or TCP</i> |
|--------------------|---------------------|---|-----------------|-------------------|
| 21 | FTP | Provisioning, Logs | | TCP |
| 22 | SSH | Admin | Admin | TCP |
| 23 | Telnet ¹ | Admin | | TCP |
| 53 | DNS | | | UDP |
| 67 | DHCP | Server | | UDP |
| 68 | DHCP | Client | | UDP |
| 69 | TFTP | Provisioning, Logs | | UDP |
| 80 | HTTP | Provisioning, Logs, Pull
Web interface, Poll | | TCP |

| <i>Port Number</i> | <i>Protocol</i> | <i>Outgoing</i> | <i>Incoming</i> | <i>UDP or TCP</i> |
|--------------------|--------------------|-------------------------|------------------------------------|-------------------|
| 123 | NTP | Time Server | | UDP |
| 389 | LDAP | | | |
| 443 | HTTPS | Provisioning, Logs | HTTP Pull Web interface, HTTP Push | TCP |
| 514 | Syslog | Logs | | |
| 636 | LDAP | | | |
| 1719 | H.323 ² | RAS Signaling | RAS Signaling | |
| 1720 | H.323 ² | Signaling | Signaling | |
| 2222 | RTP ³ | Media Packets | Media Packets | |
| 2223 | RTCP ³ | Media Packet Statistics | Media Packet Statistics | |
| 5060 | SIP | SIP signaling | SIP signaling | |
| 5061 | SIP over TLS | Secure signaling | Secure signaling | |
| 7778 | OCS | | | |
| 14394 | QBC Signaling | | QBC Server | TCP |
| 24800 | PDC | PDC Client messages | PDC Server messages | TCP |

¹ Telnet is disabled by default.

² H.323 is available on the VVX 1500 only.

³ RTP and RTCP can use any even port between 2222 and 2269 (2317 in VVX-1500), but this is configurable by setting `tcpIpApp.port.rtp.mediaPortRangeStart`.

Power and Startup Issues

The following table describes possible solutions to several power and startup issues.

Table 11-4: Troubleshooting Power and Startup Issues**The phone has power issues or the phone has no power.**

Determine if the problem is caused by the phone, the AC outlet, or the PoE switch. Do one of the following:

- Verify that no lights appear on the unit when it is powered up.
- Check if the phone is properly plugged into a functional AC outlet.
- Make sure that the phone isn't plugged into an outlet controlled by a light switch that is off.
- If plugged into a power strip, try plugging directly into a wall outlet instead.

The phone will not boot

If your phone will not boot, there may be a corrupt or invalid firmware image or configuration on the phone:

- Ensure that the provisioning server is accessible on the network and a valid software load and valid configuration files are available.
- Ensure that the phone is pointing to the provisioning server on the network.
- Reboot the phone.

Dial Pad Issues

The following table describes possible solutions to issues you may have with the dial pad.

Table 11-5: Troubleshooting Dial Pad Issues**The dial pad does not work**

If the dial pad on your phone does not respond, do one of the following:

- Check for a response from other feature keys or from the dial pad.
- Place a call to the phone from a known working telephone. Check for display updates.
- Press the Menu key followed by System Status and Server Status to check if the telephone is correctly registered to the server.
- Press the Menu key followed by System Status and Network Statistics. Scroll down to see if LAN port shows Active or Inactive.
- Check the termination at the switch or hub end of the network LAN cable. Ensure that the switch/hub port connected to the telephone is operational.

Screen and System Access Issues

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

Table 11-6: Troubleshooting Screen and System Access Issues

There is no response from feature key presses

If your phone is not in the active state, do one of the following:

- 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, for example.
- Navigate 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 (see [Rebooting the Phone](#)).

The display shows the message *Network Link is Down*

If you see this message, the LAN cable is not properly connected. Do one of the following:

- Check termination at the switch or hub (furthest end of the cable from the phone).
 - Check that the switch or hub is operational (flashing link/status lights).
 - Press Menu followed by Status > Network. Scroll down to verify that the LAN is active.
 - Ping the phone from another machine.
 - Reboot the phone to attempt re-registration to the call server (navigate to **Menu > Settings > Advanced > Reboot Phone**).
-

Calling Issues

The following table provides possible solutions to a number of generic calling issues.

Table 11-7: Troubleshooting Calling Issues

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 using in-line powering, have your system administrator check that the switch is supplying power to the phone.

The dial tone is not present on one of the audio paths

If dial tone is not present on one of the audio paths, do one of the following:

- Switch between Handset, Headset (if present) or Handsfree Speakerphone to see if dial tone is present on another path.
 - If dial tone exists on another path, connect a different handset or headset to isolate the problem.
 - Check configuration for gain levels.
-

The phone does not ring

If there is a no ring tone, 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 through the Handsfree Speakerphone.

The line icon shows an unregistered line icon

If you see one of the following icons the phone line is unregistered. Register the line and try to place a call.

Unregistered Line Icons:  (most phones)  (VFX).

Registered Line Icons:  (most phones)  (VFX).

Display Issues

The following table provides tips for resolving display screen issues.

Table 11-8: Troubleshooting Display Issues

There is no display or the display is incorrect

If there is no display, power may not be correctly supplied to the phone. Do one of the following:

- Check that the display is illuminated.
- Make sure the power is inserted properly at the rear of the phone.
- If using Power over Ethernet (PoE) powering, check that the PoE switch is supplying power to the phone.
- Use the screen capture feature to determine if the display on the phone is incorrect. See [Capturing the Phone's Current Screen](#).

The display is too dark or too light

The phone contrast may be set incorrectly. To adjust the contrast, do one of the following:

- Adjust the contrast (Refer the phone's User Guide).
- Reboot the phone to obtain the default level of contrast (see [Rebooting the Phone](#)).
- Use the screen capture feature to see if the screen displays properly in the capture. See [Capturing the Phone's Current Screen](#).

The display is flickering

Certain types of older fluorescent lighting cause the display to flicker. If your phone is in an environment lit with fluorescent lighting, do one of the following:

- Move the Polycom phone away from the lights.
 - Replace the lights.
-

The time and date are flashing

If the time in date are flashing, you have disconnected the phone from the LAN or there is no SNTP time server configured. Do one of the following (for instructions, see [Setting the Time and Date Display](#)):

- Reconnect the phone to the LAN.
 - Configure an SNTP server.
 - Disable the time and date (if you do not wish to connect your phone to a LAN or SNTP server).
-

Audio Issues

The next table briefly describes possible solutions to audio issues.

Table 11-9: Troubleshooting Audio Issues

There is no audio on the headset

If there is no audio on your headset, the connections may not be correct. Do one of the following:

- Ensure the headset is plugged into the jack marked Headset at the rear of the phone.
 - Ensure the headset amplifier (if present) is turned on and/or adjust the volume).
-

There are audio or echo issues

If you experience echo issues, see [Technical Bulletin 16249: Troubleshooting Audio and Echo Issues on SoundPoint IP Phones](#).

Licensed Feature Issues

The following table briefly explains issues and solutions involving features requiring a license.

Table 11-10: Troubleshooting Feature License Issues

Voice Quality Monitoring or H.323 is not available on the phone.

You need a license for Voice-Quality Monitoring, and H.323 support. If you cannot access any of the features, check your licenses on the phone by navigating to **Menu > Status > Licenses**.

If your phone is not installed with UC Software version 4.0.0 or later, you will also require a license for Conference Management, Corporate Directory, and Call Recording.

Upgrading Issues

The next table explains several possible solutions to issues that may occur during or after a software upgrade.

Table 11-11: Troubleshooting Software Upgrading Issues

SoundPoint IP 300, 301, 320, 330, 430, 500, 501, 600, 601, or SoundPoint IP 4000 phones behave incorrectly or do not display new features.

New features are not supported on the SoundPoint IP 300, 301, 320, 330, 430, 500, 501, 600, and 601 and SoundStation IP 4000 and/or the configuration files have not been correctly modified. These phones are not able to read the new configuration parameters, and will attempt to load the new application. See [Supporting Legacy Phones](#).

The VVX 1500 phone will not upgrade when provisioned by a Polycom CMA system

The VVX 1500 or Polycom CMA software may be out of date or incorrectly configured:

- Upgrade the VVX 1500 phone to UC Software 3.3.0 using the provisioning server. See [Upgrading Polycom UC Software](#).
- Ensure that the Polycom CMA system is running software version 5.0
- Change the appropriate CMA parameters through the phone's user interface. See [Provisioning VVX Phones Using a Polycom CMA System](#).

Certain settings or features are not working as expected on the phone

The phone's configuration may be incorrect or incompatible. Check for errors on the phone by navigating **Menu > Status > Platform > Configuration**. If there are *Errors Found*, *Unknown Params*, or *Invalid values*, correct your configuration files and restart the phone.

The phone displays a **Config file error** message for 5-seconds after it boots up (see the following figure)



Pre-UC Software 3.3.0 configuration files are being used with UC Software 3.3.0. Specifically, the following parameters are in the configuration files:

- one.chord.ringer.x.freq.1
- se.pat.callProg.x.name
- ind.anim.IP_500.x.frame.x. duration
- ind.pattern.1.step.x.state
- feature.2.name
- feature.9.name

Also the configuration files contain:

- more than 100 “unknown” parameters
- more than 100 “out-of-range” parameters
- more than 100 “invalid” parameters

Correct the configuration files, remove the invalid parameters, and restart the phone.

When upgrading phone software using the Web Configuration Utility, the phone is unable to connect to the Polycom Hosted Server

Occasionally, the phone is unable to connect to the Polycom Hosted Server because:

- The Polycom Hosted Server is temporarily unavailable.
- There isn't any software upgrade information for the phone to receive.
- The network configuration is preventing the phone from connecting to the Polycom Hosted Server.

Note: UC Software 4.0.0 does not support internet access for software upgrades through a Web proxy.

To troubleshoot the issue:

- Try upgrading your phone later.
- Verify that new software is available for your phone. To check, see the [Polycom UC Software/Polycom SIP Software Release Matrix](#).
- Verify that your network's configuration will allow the phone to connect to <http://downloads.polycom.com>.

If the issue persists, try manually upgrading your phone's software. To upgrade phone software using this method, see [Setting Up the Provisioning Server](#).

SoundStation Duo Failover Issues

The next table explains a possible solution to an issue that may occur on A SoundStation Duo conference phone.

Table 11-12: Troubleshooting SoundStation Duo Failover Issues

SoundStation Duo does not work in SIP mode

You can set up your phone so that if it is unable to operate in SIP mode, it will automatically switch over to PSTN mode.

In order for this failover to take place, you need to do the following:

- Connect your phone to an analog phone jack.
- Connect your phone to the network.
- Connect your phone to a power supply other than a Power over Ethernet (PoE) source. (Or, connect your phone to a power supply and a PoE source.)
- Configure your phone to operate in **Auto (Automatic Mode Detect)** mode.
- Register your phone to a SIP server.

If the phone is unable to register with a SIP server (including any redundant servers), or the network connection is unavailable, the phone will fail over to PSTN mode. When failover occurs, SIP features (such as forwarding, transferring, and messaging) will be unavailable.

During failover, all SIP calls will end. However, you will be able to place PSTN calls immediately, without having to restart the phone. When SIP connectivity is re-established, ongoing PSTN calls will continue. Subsequent calls you place will use SIP mode.

Note: *Your phone will only fail over from SIP to PSTN mode. It cannot fail over from PSTN to SIP mode.*

Chapter 12: Miscellaneous Maintenance Tasks

This chapter shows you how to maintain the Polycom® UC Software. This includes:

- [Trusted Certificate Authority List](#)
- [Encrypting Configuration Files](#)
- [Polycom UC Software Dependencies](#)
- [Supported VVX 1500 and CMA Server Interoperability](#)
- [Multiple Key Combinations](#)
- [Default Feature Key Layouts](#)
- [Internal Key Functions](#)
- [Assigning a VLAN ID Using DHCP](#)
- [Parsing Vendor ID Information](#)
- [Product, Model, and Part Number Mapping](#)
- [Disabling the PC Ethernet Port](#)
- [Capturing the Phone's Current Screen](#)
- [LLDP and Supported TLVs](#)

Trusted Certificate Authority List

The phone trusts the following certificate authorities by default:

- AAA Certificate Services by COMODO
- ABAecom (sub., Am. Bankers Assn.) Root CA
- Add Trust Class1 CA Root by COMODO
- Add Trust External CA Root by COMODO
- Add Trust Public CA Root by COMODO
- Add Trust Qualified CA Root by COMODO
- ANX Network CA by DST
- American Express CA
- American Express Global CA
- BelSign Object Publishing CA
- BelSign Secure Server CA

- COMODO CA Limited
- COMODO Certificate Authority
- Deutsche Telekom AG Root CA
- Digital Signature Trust Co. Global CA 1
- Digital Signature Trust Co. Global CA 2
- Digital Signature Trust Co. Global CA 3
- Digital Signature Trust Co. Global CA 4
- Entrust Worldwide by DST
- Entrust.net Premium 2048 Secure Server CA
- Entrust.net Secure Personal CA
- Entrust.net Secure Server CA
- Equifax Premium CA
- Equifax Secure CA
- Equifax Secure eBusiness CA 1
- Equifax Secure eBusiness CA 2
- Equifax Secure Global eBusiness CA 1
- GeoTrust Primary Certification Authority
- GeoTrust Global CA
- GeoTrust Global CA 2
- GeoTrust Universal CA
- GeoTrust Universal CA 2
- GTE CyberTrust Global Root
- GTE CyberTrust Japan Root CA
- GTE CyberTrust Japan Secure Server CA
- GTE CyberTrust Root 2
- GTE CyberTrust Root 3
- GTE CyberTrust Root 4
- GTE CyberTrust Root 5
- GTE CyberTrust Root CA
- GlobalSign Partners CA
- GlobalSign Primary Class 1 CA
- GlobalSign Primary Class 2 CA
- GlobalSign Primary Class 3 CA

-
- GlobalSign Root CA
 - Go Daddy Class 2 Certification Authority Root Certificate
 - Go Daddy Class 2 Certification Authority Root Certificate – G2
 - National Retail Federation by DST
 - RSA 2048 v3 Root CA
 - Secure Certificate Services by COMODO
 - TC TrustCenter, Germany, Class 1 CA
 - TC TrustCenter, Germany, Class 2 CA
 - TC TrustCenter, Germany, Class 3 CA
 - TC TrustCenter, Germany, Class 4 CA
 - Thawte Personal Basic CA
 - Thawte Personal Freemail CA
 - Thawte Personal Premium CA
 - Thawte Premium Server CA
 - Thawte Server CA
 - Thawte Universal CA Root
 - Trusted Certificate Services by COMODO
 - UTN-DATA Corp SGC by COMODO
 - UTN-USER First-Client Authentication and Email by COMODO
 - UTN-USER First-Hardware by COMODO
 - UTN-USER First-Object by COMODO
 - UPS Document Exchange by DST
 - ValiCert Class 1 VA
 - ValiCert Class 2 VA
 - ValiCert Class 3 VA
 - Verisign 2048 Root CA
 - VeriSign Class 4 Primary CA
 - Verisign Class 1 Public Primary Certification Authority
 - Verisign Class 1 Public Primary Certification Authority - G2
 - Verisign Class 1 Public Primary Certification Authority - G3
 - Verisign Class 2 Public Primary Certification Authority
 - Verisign Class 2 Public Primary Certification Authority - G2
 - Verisign Class 2 Public Primary Certification Authority - G3

- Verisign Class 3 Public Primary Certification Authority
- Verisign Class 3 Public Primary Certification Authority - G2
- Verisign Class 3 Public Primary Certification Authority - G3
- Verisign Class 3 Public Primary Certification Authority – G5
- Verisign Class 4 Public Primary Certification Authority - G2
- Verisign Class 4 Public Primary Certification Authority - G3
- Verisign/RSA Commercial CA
- Verisign/RSA Secure Server CA
- Windows Root Update by COMODO



Troubleshooting: My Certificate Authority is Not Listed

Polycom endeavors to maintain a built-in list of the most commonly used CA Certificates. Due to memory constraints, we cannot ensure a complete set of certificates. If you are using a certificate from a commercial Certificate Authority not in the list above, you may [submit a Feature Request](#) for Polycom to add your CA to the trusted list. At this point, you can use the Custom Certificate method to load your particular CA certificate into the phone. Refer to [Using Custom Certificates on Polycom Phones \(Technical Bulletin 17877\)](#).

Encrypting Configuration Files

The phone can recognize encrypted files. Phones can download encrypted files from the provisioning server and can encrypt files before uploading them to the provisioning server. There must be an encryption key on the phone to perform these operations. You can encrypt configuration files (excluding the master configuration file), contact directories, and configuration override files.

You can generate your own 32 hex-digit, 128 bit key or use Polycom's Software Development Kit (SDK) to generate a key and to encrypt and decrypt configuration files on a UNIX or Linux server. The SDK is distributed as source code that runs under the UNIX operating system.



Web Info: Using the SDK to Encrypt Configuration Files

To request the SDK and quickly install the generated key, see [Quick Tip 67442: When Encrypting Polycom UC Software Configuration Files](#).

The SDK generates a random key and applies Advanced Encryption Standard (AES) 128 in Cipher Block Chaining (CBC) mode. For example, a key can look like this:

```
Crypt=1;KeyDesc=companyNameKey1;Key=06a9214036b8a15b512e03d53412006;
```

The `device.set`, `device.sec.configEncryption.key`, and `device.sec.configEncryption.key.set` configuration file parameters are used to set the key on the phone.

If the phone doesn't have a key, it must be downloaded to the phone in plain text (a potential security concern if not using HTTPS). If the phone already has a key, a new key can be downloaded to the phone encrypted using the old key (see [below](#)).

Polycom recommends that you give each key a unique descriptive string in order to identify which key was used to encrypt a file. This makes provisioning server management easier.

After encrypting a configuration file, it is useful to rename the file to avoid confusing it with the original version, for example rename **site.cfg** to **site.enc**. However, the directory and override filenames cannot be changed in this manner.



Troubleshooting: My Phone Keeps Displaying an Error Message for My Encrypted File

If a phone downloads an encrypted file that it cannot decrypt, the action is logged, and an error message displays. The phone will continue to do this until the provisioning server provides an encrypted file that can be read, an unencrypted file, or the file is removed from the master configuration file list.

To check whether an encrypted file is the same as an unencrypted file:

- 1 Run the `configFileEncrypt` utility (available from Polycom Support) on the unencrypted file with the "-d" option. This shows the "digest" field.
- 2 Look at the encrypted file using text editor and check the first line that shows a "Digest=..." field. If the two fields are the same, then the encrypted and unencrypted file are the same.

For security purposes, you can change the key on the phones and the server from time to time.

To change a key on the phone:

- 1 Put all encrypted configuration files on the provisioning server to use the new key.
The phone may reboot multiple times.
The files on the server must be updated to the new key or they must be made available in unencrypted format. Updating to the new key requires decrypting the file with the old key, then encrypting it with the new key.
- 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 Provisioning the phone again so that it will download the new key. The phone will automatically reboot a second time to use the new key.

Note that configuration files, contact directory files and configuration override files may all need to be updated if they were already encrypted. In the case of configuration override files, they can be deleted from the provisioning server so that the phone will replace them when it successfully boots.

Polycom UC Software Dependencies

Notwithstanding the hardware backward compatibility mandate, there have been times throughout the life of Polycom® phones when certain dependencies on specific Updater (previously known as the BootROM) and UC Software (previously known as SIP software) versions have been necessitated.

Use [Table 12-1: UC Software Dependencies](#) to view some of the major dependencies that you are likely to encounter.

Table 12-1: UC Software Dependencies

| <i>Model</i> | <i>Updater/BootROM</i> | <i>UC Software/SIP Software</i> |
|---|------------------------|---------------------------------|
| IP 321/331 | Not applicable | 4.0.0 or later |
| IP 335 | Not applicable | 4.0.0 or later |
| IP 450 | Not applicable | 4.0.0 or later |
| IP 550 ¹ | Not applicable | 4.0.0 or later |
| IP 560 ¹ | Not applicable | 4.0.0 or later |
| IP 650/EM ¹ | Not applicable | 4.0.0 or later |
| IP 650/BEM | Not applicable | 4.0.0 or later |
| IP 5000 | Not applicable | 4.0.0 or later |
| IP 6000 | Not applicable | 4.0.0 or later |
| Duo ⁵ | Not applicable | UC Software 4.0.0B or later |
| VVX 500 | Not applicable | UC Software 4.0.1 or later |
| VVX 1500 ³ | 4.1.4 or later | 3.2.2 or later |
| SpectraLink 8400 Series Handsets ⁴ | Not applicable | UC Software 4.0.1 or later |

¹ SoundPoint IP 550, 560, and 650 phones manufactured as of February 2009 have additional BootROM/SIP software dependencies. For more information, refer to [Technical Bulletin TB 46440: Notice of Product Shipping Configuration Change](#).

² As of SIP 3.2.2, the BootROM 4.1.4 software is contained within the software distribution. You cannot downgrade to pre-SIP 3.2 software versions.

³ The SpectraLink 8400 Series handsets are supported as of UC Software 4.0.0.

⁴ The SoundStation Duo conference phone is supported as of UC Software 4.0.0B.

In addition to the Updater/BootROM and UC/SIP Software dependencies, there are certain restrictions with regard to upgrading or downgrading from one Updater release to another Updater release. These restrictions are typically caused by the addition of features that change the way Updater provisioning is done, so the older version becomes incompatible. This is true for upgrading to UC Software 4.0.0, which contains the Updater software, and downgrading from UC Software 4.0.0.



Web Info: Upgrading to UC Software 4.0.0

For detailed information on upgrading to UC Software 4.0.0, see [Technical Bulletin 64731: Polycom® UC Software 4.0.0: Upgrade and Downgrade Methods](#).

There is always a way to move forward with Updater/BootROM releases, although it may be a two- or three-step procedure sometimes. There are cases where it is impossible to move backward. Make special note of these cases before upgrading. For the latest information, refer to the latest *UC Software Release Notes* on the [Polycom UC Software Support Center](#).

Supported VVX 1500 and CMA Server Interoperability

To operate your VVX 1500 phone with the CMA Server, Polycom recommends that you review the latest *UC Software Release Notes*, available on the [Polycom UC Software/Polycom SIP Software Release Matrix](#) for the appropriate VVX 1500 phone and Polycom® Converged Management Application™ (CMA™) system.

Multiple Key Combinations

You can use multiple key combinations on your Polycom phones to reboot the phone, to restore the phone to factory default values, or to upload log files from the phone to your provisioning server.



Web Info: Resetting and Rebooting Your Phone

For other methods for resetting and rebooting your Polycom phones, refer to [Quick Tip 18298: Updating, Troubleshooting, and Resetting SoundPoint IP, SoundStation IP, and VVX 1500 Phones](#).

Rebooting the Phone

Rebooting the phone downloads new software and new configuration files if they exist on the provisioning server.



Timesaver: Download New Configuration Files Without Rebooting Your Phone

As of UC Software 3.3.0, not all configuration parameter changes require the phone to restart or reboot. You can update your phone's configuration by navigating to **Menu > Settings > Basic** and selecting **Update Configuration**. If there is new software on the provisioning server, the phone will restart or reboot to download the software. If there are configuration file changes, your phone will only restart if it is necessary. Otherwise, the phone will download the new configuration files without restarting.

You can use a multiple key combination to reboot your phone. Depending on your phone model, press and hold the following keys simultaneously until you hear a confirmation tone (for about three seconds):

- IP 321, 331, 335: Volume-, Volume+, Hold, and Handsfree
- IP 450, 550, 560, and 650: Volume-, Volume+, Mute, and Messages
- IP 6000: *, #, Volume+, and Select
- IP 5000, Duo: *, #, Volume-, and Volume+
- VVX 1500: Delete, Volume-, Volume+, and Select
- VVX 500 and SpectraLink 8400 Series: 0, 1, and 3 dial pad keys.



Power Tip: Quickly Restarting Your Phone

As of SIP 3.2.0, users can restart their phones by pressing the **Menu** key and selecting **Settings > Basic > Restart Phone**. If new Updater or Polycom UC Software is available on the provisioning server, the phone will download the software when it restarts.

Resetting to Factory Defaults

Resetting the phone to factory defaults clears the flash parameters and removes log files, user data, and cached data.

You can use a multiple key combination to reset your phone to the factory defaults. Depending on your phone model, press and hold the following keys simultaneously during the Updater/BootROM countdown process until the password prompt displays:

- SoundPoint IP 550, 560, and 650, and VVX 1500: 4, 6, 8, and * dial pad keys
- SoundPoint IP 321, 331, 335, 450, SoundStation 5000 and Duo: 1, 3, 5, and 7 dial pad keys

- SoundStation IP 6000: 6, 8, and * dial pad keys
- VVX 500 and SpectraLink 8400 Series: 1, 3, and 5 dial pad keys



Tip: Old Reset Behavior

Before UC Software 4.0.0, this multiple key combination performed a device reset only, clearing the flash parameters and deleting all log files. Within the Updater, this is still true.

Enter the administrator password to initiate the reset. Resetting to factory defaults will also reset the administrator password (factory default password is 456). Polycom recommends that you change the administrative password from the default value.



Settings: Resetting a VVX 1500 D to Default Values will Disable the H.323 Protocol

After you reset to factory defaults on a Polycom VVX 1500 D phone, you must re-enable the H.323 protocol (through a configuration file change or using Web Configuration Utility). See [H.323 Protocol](#).

Updating Log Files

Uploading the log files copies the log files from the phone to the provisioning server. The files called **<MACaddress>-now-xxx.log** are created.

You can use a multiple key press to upload log files to your provisioning server. Depending on your phone model, press and hold one the following keys simultaneously until you hear a confirmation tone (for about three seconds):

- SoundPoint IP 321, 331, 335: Menu, Dial, and the two Line keys
- SoundPoint IP 450, 550, 560, 650, and VVX 1500: Up, Down, Left, and Right arrow keys
- SoundStation 5000 and Duo: Up, Down, Left, and Right arrow keys and Select key
- SoundStation IP 6000: Menu, Exit, Off-hook/Handsfree, Redial
- VVX 500 and SpectraLink 8400 Series: 1, 5, and 9 dial pad keys

Setting Base Profile

Setting the base profile allows for quick setup of Polycom phones with Microsoft Lync Server 2010.

You can use a multiple key combination to set the base profile on a particular Polycom phone. Depending on your phone model, press and hold the following keys simultaneously for about three seconds until you hear a confirmation tone:

- IP 321, 331, 335, 450, 5000, 6000, Duo: 1, 2, 4, and 5 dial pad keys

- IP 550, 560, and 650: 5, 7, 8, and * dial pad keys
- VVX 500 and 1500 and SpectraLink 8400 Series: 1, 4, and 9 dial pad keys

A login screen displays. Enter the administrator password (default 456) to initiate the setup. Polycom recommends that you change the administrative password from the default value.

Default Feature Key Layouts

The following figures and tables show the default key layouts for the Polycom SoundPoint IP 321/331/335, 450, 550, 560, and 650 desktop phones, the SoundStation IP 5000, and 6000 conference phones, the SoundStation Duo conference phone, the VVX 500 and 1500 business media phone, and the SpectraLink 8400 Series wireless handsets.

The illustration of the SoundPoint IP 321/331/335 series phone is followed by [Table 12-2: SoundPoint IP 321, 331, and 331 Phone Key Functions](#), which shows the available phone key functions.

SoundPoint IP 321/331/335

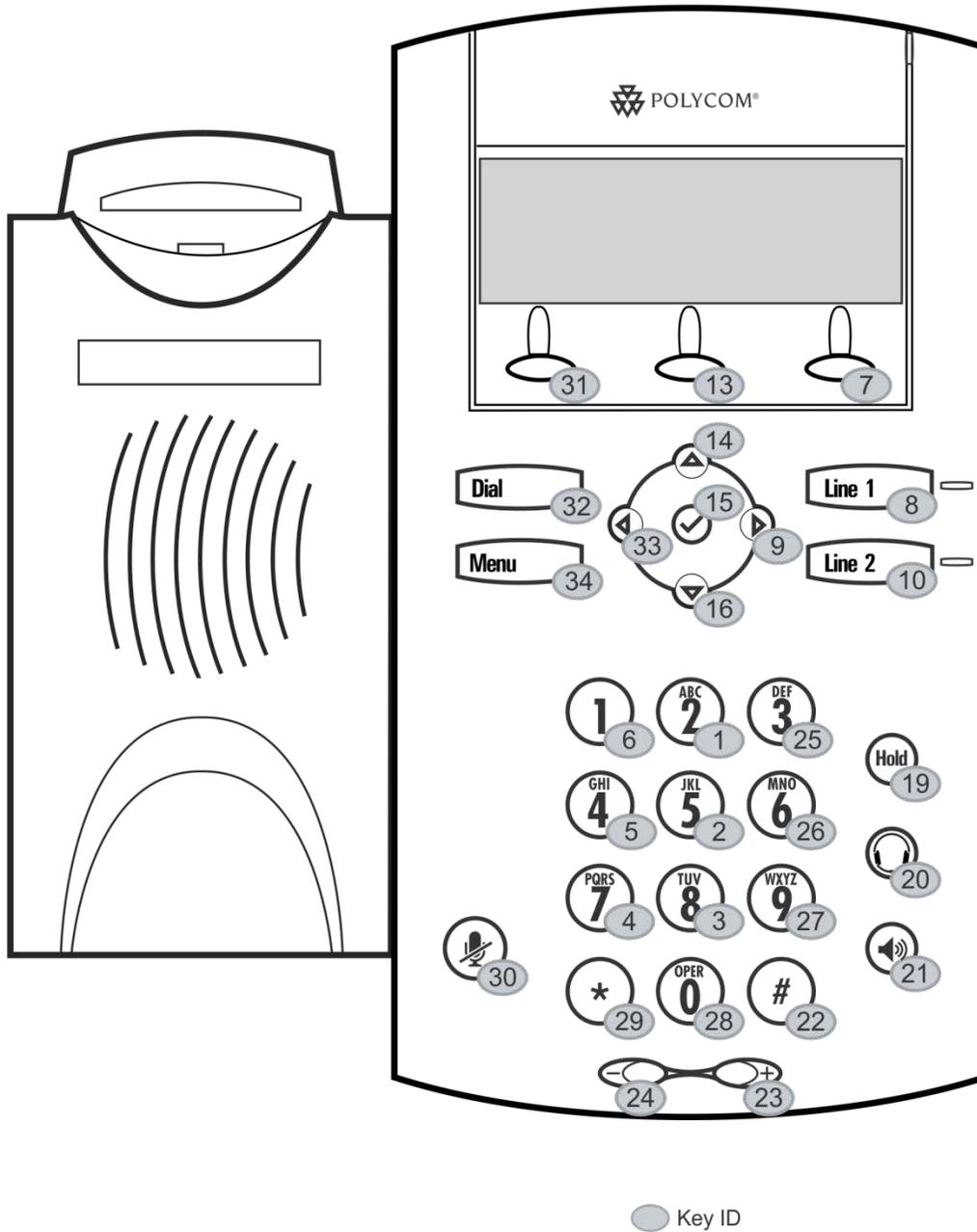


Table 12-2: SoundPoint IP 321, 331, and 331 Phone Key Functions

| Key | Function | Key | Function | Key | Function | Key | Function |
|-----|----------|-----|----------|-----|----------|-----|----------|
| 1 | Dialpad2 | 12 | n/a | 23 | VolUp | 34 | Menu |
| 2 | Dialpad5 | 13 | SoftKey2 | 24 | VolDown | 35 | n/a |
| 3 | Dialpad8 | 14 | ArrowUp | 25 | Dialpad3 | 36 | n/a |

| <i>Key</i> | <i>Function</i> | <i>Key</i> | <i>Function</i> | <i>Key</i> | <i>Function</i> | <i>Key</i> | <i>Function</i> |
|------------|-----------------|------------|-----------------|------------|-----------------|------------|-----------------|
| 4 | Dialpad7 | 15 | Select | 26 | Dialpad6 | 37 | n/a |
| 5 | Dialpad4 | 16 | ArrowDown | 27 | Dialpad9 | 38 | n/a |
| 6 | Dialpad1 | 17 | n/a | 28 | Dialpad0 | 39 | n/a |
| 7 | SoftKey3 | 18 | n/a | 29 | DialpadStar | 40 | n/a |
| 8 | Line1 | 19 | Hold | 30 | MicMute | 41 | n/a |
| 9 | ArrowRight | 20 | Headset | 31 | SoftKey1 | 42 | n/a |
| 10 | Line2 | 21 | Handsfree | 32 | Dial | | |
| 11 | n/a | 22 | DialpadPound | 33 | ArrowLeft | | |

The illustration of the SoundPoint IP 450 is followed by [Table 12-3: SoundPoint IP 450 Phone Key Functions](#), which shows the available phone key functions.

SoundPoint IP 450

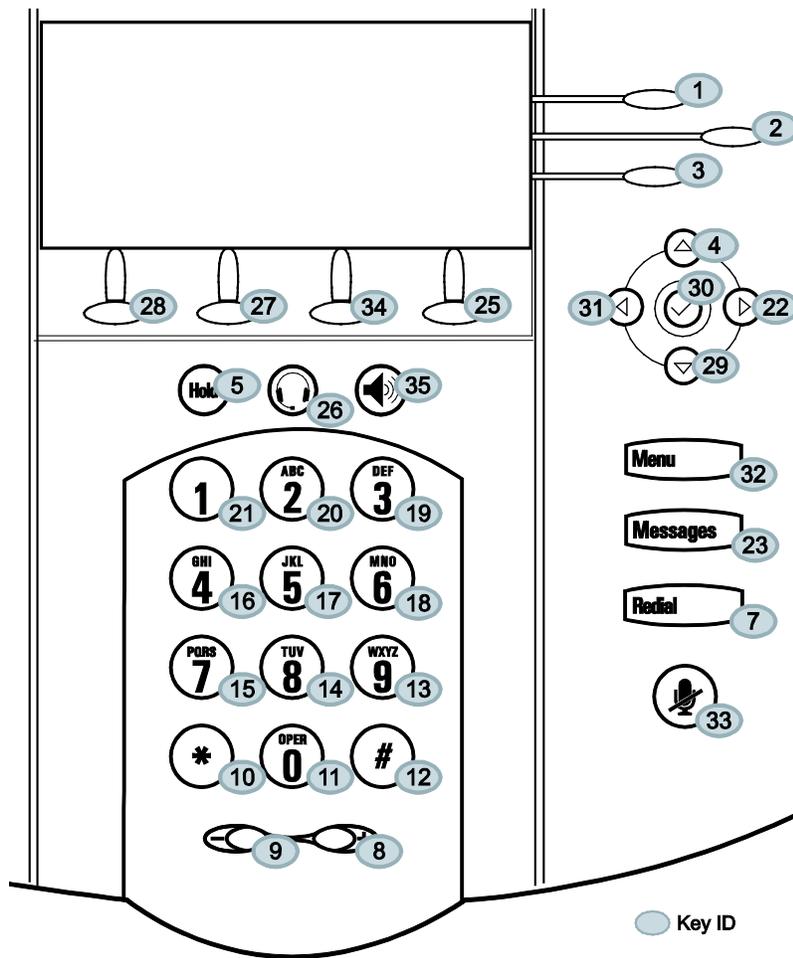


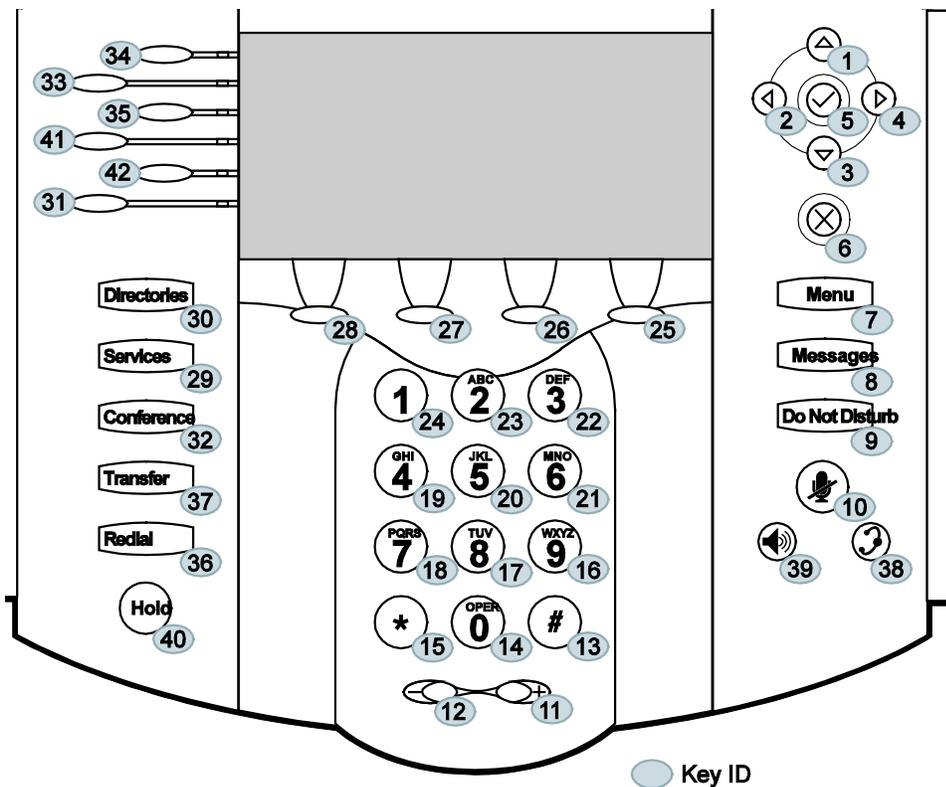
Table 12-3: SoundPoint IP 450 Phone Key Functions

| Key | Function | Key | Function | Key | Function | Key | Function |
|-----|----------|-----|--------------|-----|-----------|-----|-----------|
| 1 | Line1 | 12 | DialpadPound | 23 | Messages | 34 | SoftKey3 |
| 2 | Line2 | 13 | Dialpad9 | 24 | n/a | 35 | Handsfree |
| 3 | Line3 | 14 | Dialpad8 | 25 | Softkey4 | 36 | n/a |
| 4 | ArrowUp | 15 | Dialpad7 | 26 | Headset | 37 | n/a |
| 5 | Hold | 16 | Dialpad4 | 27 | SoftKey2 | 38 | n/a |
| 6 | n/a | 17 | Dialpad5 | 28 | SoftKey1 | 39 | n/a |
| 7 | Redial | 18 | Dialpad6 | 29 | ArrowDown | 40 | n/a |
| 8 | VolUp | 19 | Dialpad3 | 30 | Select | 41 | n/a |

| Key | Function | Key | Function | Key | Function | Key | Function |
|-----|-------------|-----|------------|-----|-----------|-----|----------|
| 9 | VolDown | 20 | Dialpad2 | 31 | ArrowLeft | 42 | n/a |
| 10 | DialpadStar | 21 | Dialpad1 | 32 | Menu | | |
| 11 | Dialpad0 | 22 | ArrowRight | 33 | MicMute | | |

The illustration of the SoundPoint IP 550/560/650 is followed by [Table 12-4: SoundPoint IP 550, 560, and 650 Phone Key Functions](#), which shows the available phone key functions.

SoundPoint IP 550/560/650



Note: Differences Between SoundPoint IP 550/560 and SoundPoint IP 650 Line Keys

The SoundPoint IP 550 and 560 have only the top four lines keys. Key IDs 31 and 42 are not used on SoundPoint IP 550 and 560 phones.

Table 12-4: SoundPoint IP 550, 560, and 650 Phone Key Functions

| Key | Function | Key | Function | Key | Function | Key | Function |
|-----|--------------|-----|--------------|-----|--------------|-----|-----------|
| 1 | ArrowUp | 12 | VolDown | 23 | Dialpad2 | 34 | Line1 |
| 2 | ArrowLeft | 13 | DialpadPound | 24 | Dialpad1 | 35 | Line3 |
| 3 | ArrowDown | 14 | Dialpad0 | 25 | SoftKey4 | 36 | Redial |
| 4 | ArrowRight | 15 | DialpadStar | 26 | SoftKey3 | 37 | Transfer |
| 5 | Select | 16 | Dialpad9 | 27 | SoftKey2 | 38 | Headset |
| 6 | Delete | 17 | Dialpad8 | 28 | SoftKey1 | 39 | Handsfree |
| 7 | Menu | 18 | Dialpad7 | 29 | Applications | 40 | Hold |
| 8 | Messages | 19 | Dialpad4 | 30 | Directories | 41 | Line4 |
| 9 | DoNotDisturb | 20 | Dialpad5 | 31 | Line6 | 42 | Line5 |
| 10 | MicMute | 21 | Dialpad6 | 32 | Conference | | |
| 11 | VolUp | 22 | Dialpad3 | 33 | Line2 | | |

The illustration of the SoundPoint IP 5000 is followed by [Table 12-5: SoundStation IP 5000 Phone Key Functions](#), which shows the available phone key functions.

SoundStation IP 5000

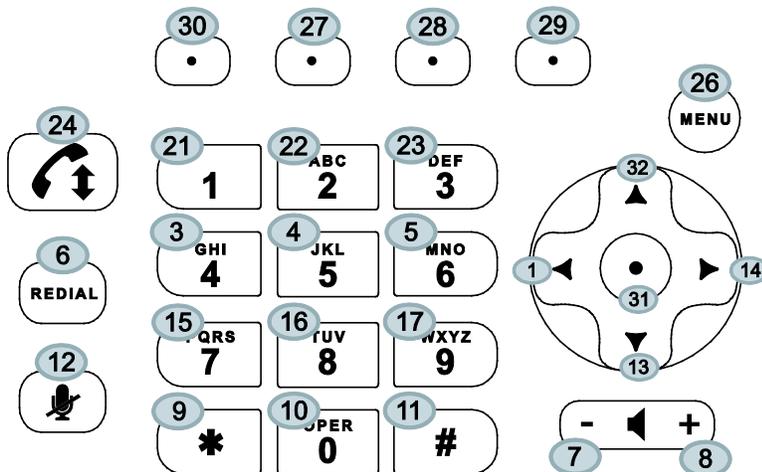


Table 12-5: SoundStation IP 5000 Phone Key Functions

| Key | Function | Key | Function | Key | Function | Key | Function |
|-----|--------------|-----|------------|-----|-----------|-----|----------|
| 1 | ArrowLeft | 12 | MicMute | 23 | Dialpad3 | 34 | n/a |
| 2 | n/a | 13 | ArrowDown | 24 | Handsfree | 35 | n/a |
| 3 | Dialpad4 | 14 | ArrowRight | 25 | n/a | 36 | n/a |
| 4 | Dialpad5 | 15 | Dialpad7 | 26 | Menu | 37 | n/a |
| 5 | Dialpad6 | 16 | Dialpad8 | 27 | SoftKey2 | 38 | n/a |
| 6 | Redial | 17 | Dialpad9 | 28 | SoftKey3 | 39 | n/a |
| 7 | VolDown | 18 | n/a | 29 | SoftKey4 | 40 | n/a |
| 8 | VolUp | 19 | n/a | 30 | SoftKey1 | 41 | n/a |
| 9 | DialpadStar | 20 | n/a | 31 | Select | 42 | n/a |
| 10 | Dialpad0 | 21 | Dialpad1 | 32 | ArrowUp | | |
| 11 | DialpadPound | 22 | Dialpad2 | 33 | n/a | | |

The illustration of the SoundPoint IP 6000 is followed by [Table 12-6: SoundPoint IP 6000 Phone Key Functions](#), which shows the available phone key functions.

SoundStation IP 6000

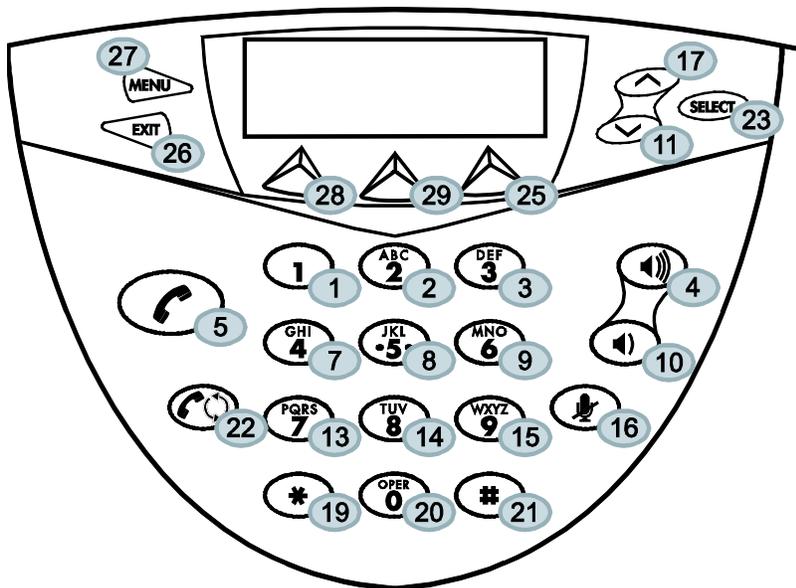


Table 12-6: SoundPoint IP 6000 Phone Key Functions

| <i>Key</i> | <i>Function</i> | <i>Key</i> | <i>Function</i> | <i>Key</i> | <i>Function</i> | <i>Key</i> | <i>Function</i> |
|------------|-----------------|------------|-----------------|------------|-----------------|------------|-----------------|
| 1 | Dialpad1 | 12 | n/a | 23 | Select | 34 | n/a |
| 2 | Dialpad2 | 13 | Dialpad7 | 24 | n/a | 35 | n/a |
| 3 | Dialpad3 | 14 | Dialpad8 | 25 | SoftKey3 | 36 | n/a |
| 4 | VolUp | 15 | Dialpad9 | 26 | Exit | 37 | n/a |
| 5 | Handsfree | 16 | MicMute | 27 | Menu | 38 | n/a |
| 6 | n/a | 17 | ArrowUp | 28 | SoftKey1 | 39 | n/a |
| 7 | Dialpad4 | 18 | n/a | 29 | SoftKey2 | 40 | n/a |
| 8 | Dialpad5 | 19 | DialpadStar | 30 | n/a | 41 | n/a |
| 9 | Dialpad6 | 20 | Dialpad0 | 31 | n/a | 42 | n/a |
| 10 | VolDown | 21 | DialpadPound | 32 | n/a | | |
| 11 | ArrowDown | 22 | Redial | 33 | n/a | | |

The illustration of the SoundStation Duo is followed by [Table 12-7: SoundStation Duo Phone Key Functions](#), which shows the available phone key functions.

SoundStation Duo

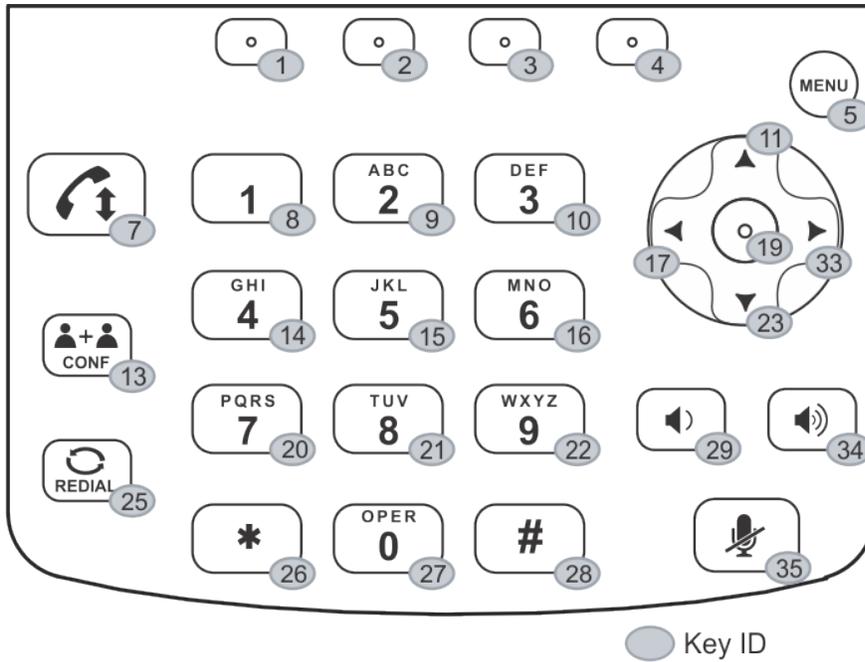


Table 12-7: SoundStation Duo Phone Key Functions

| Key | Function | Key | Function | Key | Function | Key | Function |
|-----|-----------|-----|------------|-----|--------------|-----|----------|
| 1 | SoftKey1 | 12 | n/a | 23 | ArrowDown | 34 | VolUp |
| 2 | SoftKey2 | 13 | Conference | 24 | n/a | 35 | MicMute |
| 3 | SoftKey3 | 14 | Dialpad4 | 25 | Redial | 36 | n/a |
| 4 | SoftKey4 | 15 | Dialpad5 | 26 | DialpadStar | 37 | n/a |
| 5 | Menu | 16 | Dialpad6 | 27 | Dialpad0 | 38 | n/a |
| 6 | n/a | 17 | ArrowLeft | 28 | DialpadPound | 39 | n/a |
| 7 | Handsfree | 18 | n/a | 29 | VolDown | 40 | n/a |
| 8 | Dialpad1 | 19 | Select | 30 | n/a | 41 | n/a |
| 9 | Dialpad2 | 20 | Dialpad7 | 31 | n/a | 42 | n/a |
| 10 | Dialpad3 | 21 | Dialpad8 | 32 | n/a | | |
| 11 | ArrowUp | 22 | Dialpad9 | 33 | ArrowRight | | |

The illustration of the VVX 500 is followed by [Table 12-8: VVX 500 Phone Key Functions](#), which shows the available phone key functions.



Table 12-8: VVX 500 Phone Key Functions

| Key ID | Function | Key ID | Function | Key ID | Function | Key ID | Function |
|--------|----------|--------|----------|--------|--------------|--------|----------|
| 1 | Dialpad1 | 12 | Headset | 23 | Dialpad0 | 34 | n/a |
| 2 | Dialpad2 | 13 | n/a | 24 | DialpadPound | 35 | n/a |
| 3 | VolDown | 14 | n/a | 25 | n/a | 36 | n/a |
| 4 | VolUp | 15 | Dialpad7 | 26 | Home | 37 | n/a |
| 5 | Dialpad3 | 16 | Dialpad8 | 27 | n/a | 38 | n/a |
| 6 | n/a | 17 | Dialpad9 | 28 | n/a | 39 | n/a |
| 7 | n/a | 18 | MicMute | 29 | n/a | 40 | Dialpad6 |

| Key ID | Function | Key ID | Function | Key ID | Function | Key ID | Function |
|--------|-----------|--------|-------------|--------|----------|--------|----------|
| 8 | Dialpad4 | 19 | n/a | 30 | n/a | 41 | n/a |
| 9 | Dialpad5 | 20 | n/a | 31 | n/a | 42 | n/a |
| 10 | n/a | 21 | n/a | 32 | n/a | | |
| 11 | Handsfree | 22 | DialpadStar | 33 | n/a | | |

The illustration of the VVX 1500 is followed by [Table 12-9: VVX 1500 Phone Key Functions](#), which shows the available phone key functions.

VVX 1500

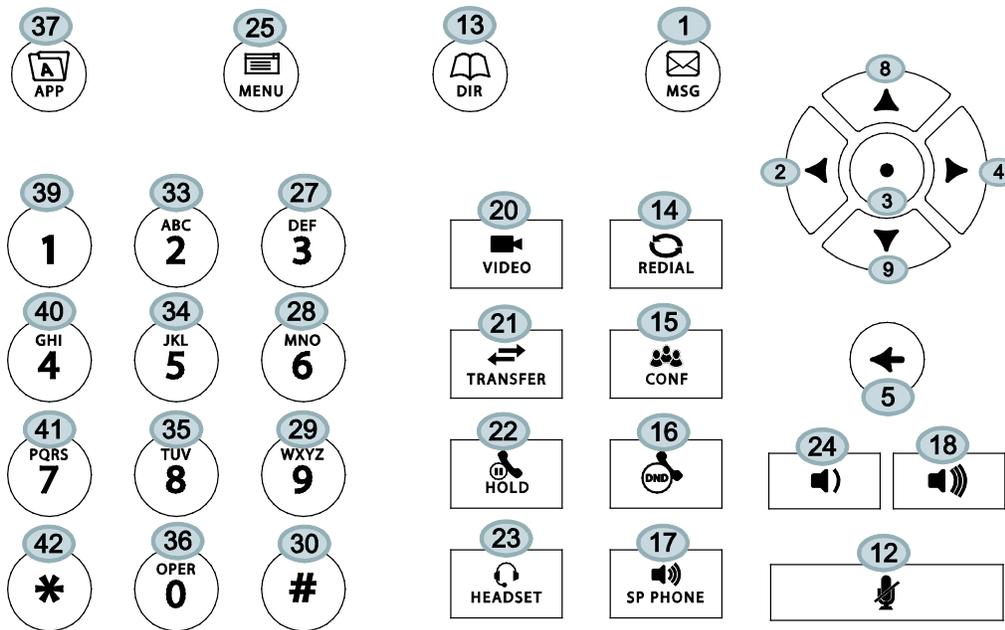


Table 12-9: VVX 1500 Phone Key Functions

| Key | Function | Key | Function | Key | Function | Key | Function |
|-----|------------|-----|--------------|-----|----------|-----|--------------|
| 1 | Messages | 12 | MicMute | 23 | Headset | 34 | Dialpad5 |
| 2 | ArrowLeft | 13 | Directories | 24 | VoIDown | 35 | Dialpad8 |
| 3 | Select | 14 | Redial | 25 | Menu | 36 | Dialpad0 |
| 4 | ArrowRight | 15 | Conference | 26 | n/a | 37 | Applications |
| 5 | Delete | 16 | DoNotDisturb | 27 | Dialpad3 | 38 | n/a |

| Key | Function | Key | Function | Key | Function | Key | Function |
|-----|-----------|-----|-----------|-----|-------------|-----|-------------|
| 6 | n/a | 17 | Handsfree | 28 | Dialpad6 | 39 | Dialpad1 |
| 7 | n/a | 18 | VolUp | 29 | Dialpad9 | 40 | Dialpad4 |
| 8 | ArrowUp | 19 | n/a | 30 | DialpadPoun | 41 | Dialpad7 |
| 9 | ArrowDown | 20 | Video | 31 | n/a | 42 | DialpadStar |
| 10 | n/a | 21 | Transfer | 32 | n/a | | |
| 11 | n/a | 22 | Hold | 33 | Dialpad2 | | |

The illustration of the SpectraLink handsets is followed by [Table 12-10: SpectraLink 8440 and 8450 Handset Key Functions](#), which shows the available phone key functions.

SpectraLink 8400 Series

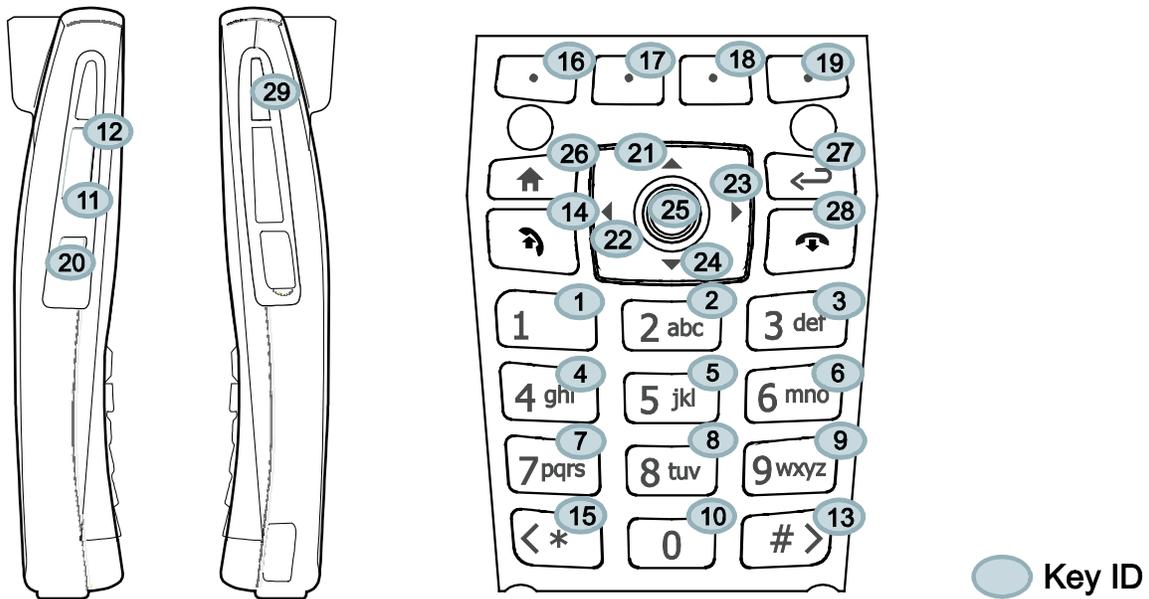


Table 12-10: SpectraLink 8440 and 8450 Handset Key Functions

| Key | Function | Key | Function | Key | Function | Key | Function |
|-----|----------|-----|--------------|-----|------------|-----|----------|
| 1 | Dialpad1 | 12 | VolDown | 23 | ArrowRight | 34 | n/a |
| 2 | Dialpad2 | 13 | DialpadPound | 24 | ArrowDown | 35 | n/a |
| 3 | Dialpad3 | 14 | Green | 25 | Select | 36 | n/a |

| <i>Key</i> | <i>Function</i> | <i>Key</i> | <i>Function</i> | <i>Key</i> | <i>Function</i> | <i>Key</i> | <i>Function</i> |
|------------|-----------------|------------|-----------------|------------|-----------------|------------|-----------------|
| 4 | Dialpad4 | 15 | DialpadStar | 26 | Home | 37 | n/a |
| 5 | Dialpad5 | 16 | SoftKey1 | 27 | Back | 38 | n/a |
| 6 | Dialpad6 | 17 | SoftKey2 | 28 | Red | 39 | n/a |
| 7 | Dialpad7 | 18 | SoftKey3 | 29 | Barcode | 40 | n/a |
| 8 | Dialpad8 | 19 | SoftKey4 | 30 | n/a | 41 | n/a |
| 9 | Dialpad 9 | 20 | Talk | 31 | n/a | 42 | n/a |
| 10 | Dialpad0 | 21 | ArrowUp | 32 | n/a | | |
| 11 | VolUp | 22 | ArrowLeft | 33 | n/a | | |

Internal Key Functions

A complete list of internal key functions for enhanced feature keys and hard key mappings is shown in [Table 12-11: Key Labels and Internal Functions](#).

The following guidelines should be noted:

- The **Label** value is case sensitive.
- Some functions are dependent on call state. Generally, if the soft key displays on a call screen, the soft key function is executable. There are some exceptions on the SoundPoint IP 321/331/335 phone (because it does not display as many soft keys).
- On the SoundPoint IP 321/331/335 phone, CallPickup and ParkedPickup refer to the same function. On other phones, CallPickup refers to the soft key function that provides the menu with separate soft keys for parked pickup, directed pickup, and group pickup.
- Some functions depend on the feature being enabled. For example, BuddyStatus and MyStatus require the presence feature to be enabled.
- Hard key remappings do not require the Enhanced Feature key feature to be enabled. This includes the SpeedDial function on older platforms. On newer platforms, use line key functions.
- The table below shows only Line1 to Line6 functions. For the SoundPoint IP 650 phones with attached Expansion Modules, Line7 to Line48 functions are also supported.

Table 12-11: Key Labels and Internal Functions

| <i>Label</i> | <i>Function</i> | <i>Notes</i> |
|----------------|------------------------------------|------------------|
| ACDAvailable | ACDAvailableFromIdle | |
| ACDLogin | ACDLoginLogout | |
| ACDLogout | ACDLoginLogout | |
| ACDUnavailable | ACDAvailableFromIdle | |
| Answer | Answer | Call screen only |
| Applications | Main Browser | |
| ArrowDown | ArrowDown | |
| ArrowLeft | ArrowLeft | |
| ArrowRight | ArrowRight | |
| ArrowUp | ArrowUp | |
| BargeIn | BargeInShowAppearances,
BargeIn | Call screen only |
| BuddyStatus | Buddy Status | |
| Callers | Callers | |
| CallList | Call Lists | |
| CallPark | ParkEntry | Call screen only |
| CallPickup | CallPickupEntry | Call screen only |
| Conference | ConferenceCall | Call screen only |
| Delete | Delete | |
| Dialpad0 | Dialpad0 | |
| Dialpad1 | Dialpad1 | |
| Dialpad2 | Dialpad2 | |
| Dialpad3 | Dialpad3 | |
| Dialpad4 | Dialpad4 | |
| Dialpad5 | Dialpad5 | |
| Dialpad6 | Dialpad6 | |

| <i>Label</i> | <i>Function</i> | <i>Notes</i> |
|----------------|--------------------------------|---------------------|
| Dialpad7 | Dialpad7 | |
| Dialpad8 | Dialpad8 | |
| Dialpad9 | Dialpad9 | |
| DialpadPound | DialpadPound | |
| DialpadStar | DialpadStar | |
| DialpadURL | Dialname | Call screen only |
| DirectedPickup | DirectedPickup | Call screen only |
| Directories | Directories | |
| Divert | Forward | |
| DoNotDisturb | Do Not Disturb menu | |
| EnterRecord | enterCallRecord | Call screen only |
| Exit | Exist existing menu | Menu only |
| FLockPhone | Lock phone | |
| GroupPickup | GroupPickup | |
| Handsfree | Handsfree | |
| Headset | Headset | Desktop phones only |
| Hold | Toggle Hold | |
| Join | Join | Call screen only |
| LCR | LastCallReturn | |
| Line1 | Line Key 1 | |
| Line2 | Line Key 2 | |
| Line3 | Line Key 3 | |
| Line4 | Line Key 4 | |
| Line5 | Line Key 5 | |
| Line6 | Line Key 6 | |
| ListenMode | Turn on speaker to listen only | |
| Menu | Menu | |

| <i>Label</i> | <i>Function</i> | <i>Notes</i> |
|---------------------------|---------------------------|-----------------------|
| Messages | Messages menu | |
| MicMute | MicMute | |
| MyStatus | MyStatus | |
| NewCall | NewCall | Call screen only |
| Null | Do nothing | |
| Offline | Offline for presence | |
| Paging | Group Paging | |
| ParkedPickup | ParkedPickup | Call screen only |
| QuickSetup | Quick Setup feature | Call screen only |
| Redial | Redial | Call screen only |
| Select | Select | |
| ServerACDAgentAvailable | serverACDAgentAvailable | |
| ServerACDAgentUnavailable | serverACDAgentUnavailable | |
| ServerACDSignIn | serverACDSignIn | |
| ServerACDSignOut | serverACDSignOut | |
| Setup | Settings menu | |
| Silence | RingerSilence | Call screen only |
| SoftKey1 | SoftKey1 | |
| SoftKey2 | SoftKey2 | |
| SoftKey3 | SoftKey3 | |
| SoftKey4 | SoftKey4 | |
| SpeedDial | SpeedDial | |
| Split | Split | Call screen only |
| Talk | Push-to-Talk | |
| Transfer | Transfer | Call screen only |
| Video | Video | Polycom VVX 1500 only |
| VoIDown | VoIDown | |

| <i>Label</i> | <i>Function</i> | <i>Notes</i> |
|--------------|-----------------|--------------|
| VolUp | VolUp | |

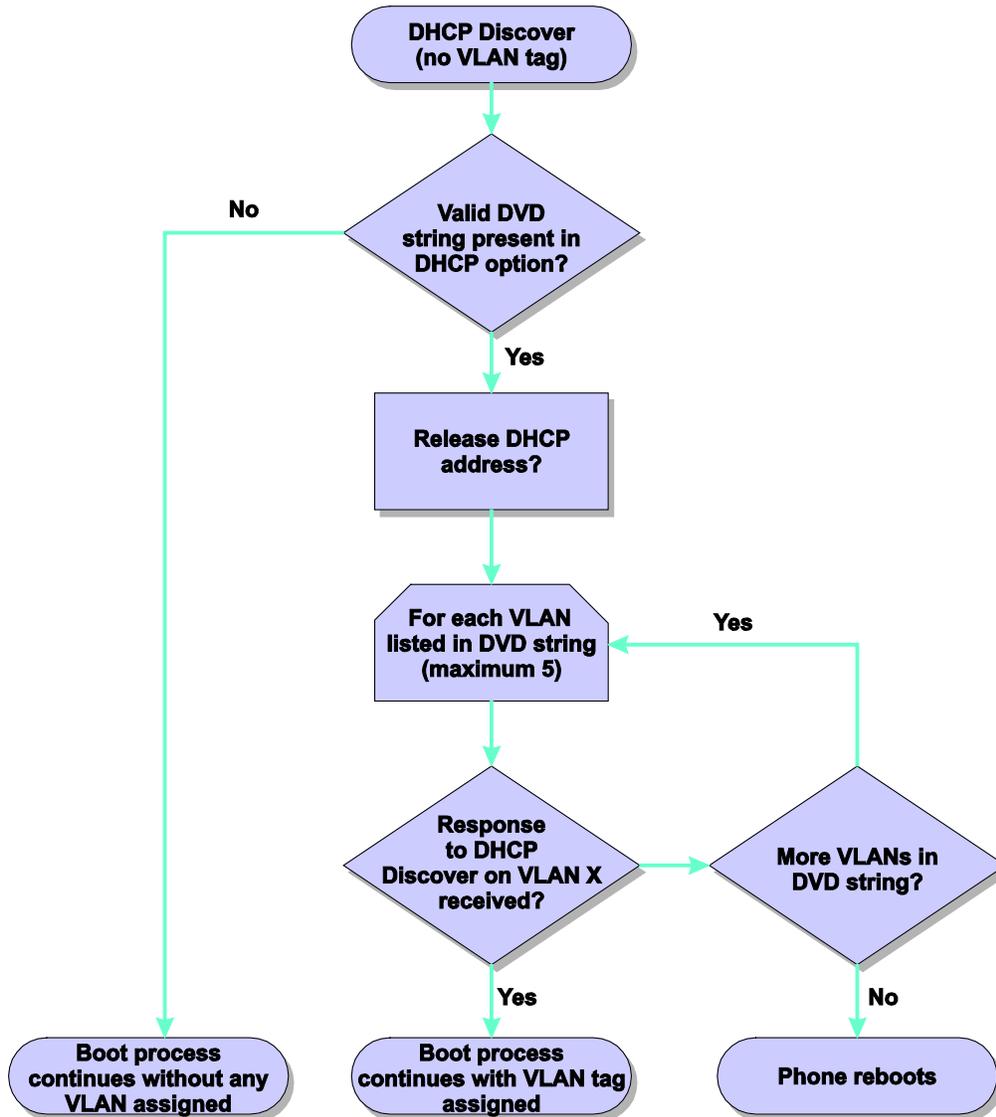
Assigning a VLAN ID Using DHCP

In deployments where it is not possible or desirable to assign a VLAN statically in the phone's network configuration menu or use CDP (Cisco Discovery Protocol) or LLDP (Link-Layer Discovery Protocol) to assign a VLAN ID, it is possible to assign a VLAN ID to the phone by distributing the VLAN ID via DHCP.

When using this method to assign the phone's VLAN ID, the phone first boots on the default VLAN (or statically configured VLAN, if first configured in the phone's network configuration menu), obtains its intended VLAN ID from the DHCP offer, then continues booting (including a subsequent DHCP sequence) on the newly obtained VLAN.

See [Figure 12-1: VLAN Using DHCP Phone Boot-Up Sequence](#) for the phone boot-up sequence when assigning a VLAN ID via DHCP.

Figure 12-1: VLAN Using DHCP Phone Boot-Up Sequence



To assign a VLAN ID to a phone using DHCP:

In the DHCP menu of the Main setup menu, set **VLAN Discovery** to **Fixed** or **Custom**.

- When set to Fixed, the phone will examine DHCP options 128,144, 157 and 191 (in that order) for a valid DVD string.
- When set to Custom, a value set in the **VLAN ID Option** will be examined for a valid DVD string.

DVD string in the DHCP option must meet the following conditions to be valid:

- Must start with "VLAN-A=" (case-sensitive)
- Must contain at least one valid ID
- VLAN IDs range from 0 to 4095

- Each VLAN ID must be separated by a “+” character
- The string must be terminated by a semi colon “;”
- All characters after the semi colon “;” will be ignored
- There must be no white space before the semi colon “;”
- VLAN IDs may be decimal, hex, or octal

The following DVD strings will result in the phone using VLAN 10:

```
VLAN-A=10;
```

```
VLAN-A=0x0a;
```

```
VLAN-A=012;
```



Note: VLAN Tags Assigned by CDP or LLDP

If a VLAN tag is assigned by CDP or LLDP, DHCP VLAN tags will be ignored.

Parsing Vendor ID Information

After the phone boots, it sends a DHCP Discover packet to the DHCP server. This is found in the Bootstrap Protocol/option ‘Vendor Class Identifier’ section of the packet and includes the phone’s part number and the BootROM version. RFC 2132 does not specify the format of this option's data, and can be defined by each vendor. To be useful, every vendor's format must be distinguishable from every other vendor's format. To make our format uniquely identifiable, the format follows RFC 3925, which uses the IANA Private Enterprise number to determine which vendor's format should be used to decode the remaining data. The private enterprise number assigned to Polycom is 13885 (0x0000363D).

This vendor ID information is not a character string, but an array of binary data.

The steps for parsing are as follows:

- 1 Check for the Polycom signature at the start of the option:
4 octet: 00 00 36 3d
- 2 Get the length of the entire list of sub-options:
1 octet
- 3 Read the field code and length of the first sub-option, 1+1 octets
- 4 If this is a field you want to parse, save the data.
- 5 Skip to the start of the next sub-option.

- 6 Repeat steps 3 to 5 until you have all the data or you encounter the End-of-Suboptions code (0xFF).

For example, the following is a sample decode of a packet from an IP 601:

```

3c 74
    o Option 60, length of Option data (part of the DHCP spec.)
00 00 36 3d
    o Polycom signature (always 4 octets)
6f
    o Length of Polycom data
01 07 50 6f 6c 79 63 6f 6d
    o sub-option 1 (company), length, "Polycom"
02 15 53 6f 75 6e 64 50 6f 69 6e 74 49 50 2d 53 50 49 50 5f 36 30 31
    o sub-option 2 (part), length, "SoundPointIP-SPIP_601"
03 10 32 33 34 35 2d 31 31 36 30 35 2d 30 30 31 2c 32
    o sub-option 3 (part number), length, "2345-11605-001,2"
04 1c 53 49 50 2f 54 69 70 2e 58 58 58 58 2f 30 38 2d 4a 75 6e 2d 30 37 20
31 30 3a 34 34
    o sub-option 4 (Application version), length, "SIP/Tip.XXXX/08-Jun-07 10:44"
05 1d 42 52 2f 33 2e 31 2e 30 2e 58 58 58 58 2f 32 38 2d 41 70 72 2d 30 35
20 31 33 3a 33 30
    o sub-option 5 (BootROM version), length, "BR/3.1.0.XXXX/28-Apr-05
13:30"
ff
    o end of sub-options

```

For the Updater, sub-option 4 and sub-option 5 will contain the same string. The string is formatted as follows:

```
<apptype>/<builddid>/<date+time>
```

where:

```
<apptype> can be 'BR' (BootROM) or 'SIP' (SIP Application)
```

Product, Model, and Part Number Mapping

In SIP 2.1.2, enhancements to the master configuration file were made to enable you to direct phone upgrades to a software image and configuration files based on a phone model number, a firmware part number, or a phone's MAC address.

The part number has precedence over the model number, which has precedence over the original version. For example, `CONFIG_FILES_2345-11560-001="phone1_2345-11560-001.cfg, sip_2345-11560-001.cfg"` will override

`CONFIG_FILES_SPIP560="phone1_SPIP560.cfg, sip_SPIP560.cfg"`, which will override `CONFIG_FILES="phone1.cfg, sip.cfg"` for a SoundPoint IP 560.

You can also add variables to the master configuration file that are replaced when the phone reboots. The variables include `PHONE_MODEL`, `PHONE_PART_NUMBER`, and `PHONE_MAC_ADDRESS`.

Use [Table 12-12: Product Name, Model Name, and Part Number](#) as a reference guide showing the product name, model name, and part number mapping for SoundPoint IP, SoundStation IP, Polycom VVX 1500, and SpectraLink 8400 Series phones.

Table 12-12: Product Name, Model Name, and Part Number

| <i>Product Name</i> | <i>Model Name</i> | <i>Part Number</i> |
|----------------------|-------------------|--|
| SoundPoint IP 300 | SPIP300 | 2345-11300-001 |
| SoundPoint IP 301 | SPIP301 | 2345-11300-010 |
| SoundPoint IP 320 | SPIP320 | 2345-12200-002
2345-12200-005 |
| SoundPoint IP 321 | SPIP321 | 2345-12360-001 |
| SoundPoint IP 330 | SPIP330 | 2345-12200-001
2345-12200-004 |
| SoundPoint IP 331 | SPIP331 | 2345-12365-001 |
| SoundPoint IP 335 | SPIP335 | 2345-12375-001 |
| SoundPoint IP 430 | SPIP430 | 2345-11402-001 |
| SoundPoint IP 450 | SPIP450 | 2345-12450-001 |
| SoundPoint IP 500 | SPIP500 | 2345-11500-001
2345-11500-010
2345-11500-020 |
| SoundPoint IP 501 | SPIP501 | 2345-11500-030
2345-11500-040 |
| SoundPoint IP 550 | SPIP550 | 2345-12500-001 |
| SoundPoint IP 560 | SPIP560 | 2345-12560-001 |
| SoundPoint IP 600 | SPIP600 | 2345-11600-001 |
| SoundPoint IP 601 | SPIP601 | 2345-11605-001 |
| SoundPoint IP 650 | SPIP650 | 2345-12600-001 |
| SoundStation IP 5000 | SSIP5000 | 3111-30900-001 |

| <i>Product Name</i> | <i>Model Name</i> | <i>Part Number</i> |
|-------------------------------|-------------------|--------------------|
| SoundStation IP 6000 | SSIP6000 | 3111-15600-001 |
| SoundStation Duo | SSDuo | 3111-19000-001 |
| SoundStructure VoIP Interface | SSTRVOIP | 3111-33215-001 |
| Polycom VVX 500 | VVX500 | 3111-44599-001 |
| Polycom VVX 1500 | VVX1500 | 2345-17960-001 |
| SpectraLink 8400 Series | SL8440 | 3111-36150-001 |
| | SL8450 | 3111-36152-001 |
| | SL8452 | 3111-36154-001 |

Disabling the PC Ethernet Port

Certain SoundPoint IP phones have a PC Ethernet port. If it is unused, it can be disabled.

The PC Ethernet port can be disabled on the SoundPoint IP 33x, 450, 550, 560, 601, and 650, and VVX 1500 through the menu (shown below). The Ethernet port can also be disabled through the configuration files.

To disable Ethernet on a supported SoundPoint IP phone:

- 1 Navigate to the phone's **Ethernet Menu (Menu > Settings > Advanced > Network Configuration > Ethernet Menu)**.

You will need to enter the administrator password to access the advanced settings, the default password is **456**.

- 2 Scroll down to **PC Port Mode** and press the **Edit** soft key.
- 3 Select **Disabled** and press the **OK** soft key.
- 4 Press the **Exit** soft key and select **Save Config**.

The phone will reboot. When the reboot is complete, the PC Ethernet port will be disabled.

Capturing the Phone's Current Screen

You can capture your phone's current screen using a Web browser.



Troubleshooting: I Can't Take a Screen Capture of the SpectraLink Site Survey Screen

You will not be able to take screen captures of the site survey screens on the SpectraLink handsets as the network connection is disabled while site survey is running.

To capture the phone's current screen:

- 1 Modify your configuration file to enable the screen capture feature.

You will need to open your configuration file in an XML editor and add the following line:

```
#comment
up
up.screenCapture.enabled 1
```

User Preferences definition

- 2 Save the configuration file and update your phone's configuration.
- 3 On the phone, turn on the screen capture feature from the **Screen Capture** menu (**Menu > Settings > Basic > Preferences > Screen Capture**).

You will need to turn the screen capture on again (repeat this step) each time the phone restarts or reboots.

- 4 In a Web browser, enter `http://<phoneIPAddress>/captureScreen` as the browser address.

To find your phone's IP address, navigate to **Menu > Status > Platform > Phone**.

The Web browser will display an image showing the phone's current screen. The image can be saved as a BMP or JPEG file.

LLDP and Supported TLVs

The Link Layer Discovery Protocol (LLDP) is a vendor-neutral Layer 2 protocol that allows a network device to advertise its identity and capabilities on the local network.

This section does not apply to the SpectraLink handsets as they do not use LLDP.



Web Info: Using the LLDP Protocol

The protocol was formally ratified as IEEE standard 802.1AB in May 2005. Refer to section 10.2.4.4 of the [LLDP-MED standard](#).

The LLDP feature (added in SIP 3.2.0) supports VLAN discovery and LLDP power management, but not power negotiation. LLDP has a higher priority than CDP and DHCP VLAN discovery.



Settings: Enabling VLAN Using Multiple Method

There are four ways to obtain VLAN on the phone and they can all be enabled, but the VLAN used is chosen by the priority of each method: 1. LLDP; 2. CDP; 3. DVD (VLAN Via DHCP); 4. Static (the VLAN ID is entered through the phone's user interface).

The following mandatory and optional Type Length Values (TLVs) are supported:

Mandatory:

- Chassis ID—Must be first TLV
- Port ID—Must be second TLV
- Time-to-live—Must be third TLV, set to 120 seconds
- End-of-LLDPDU—Must be last TLV
- LLDP-MED Capabilities
- LLDP-MED Network Policy—VLAN, L2 QoS, L3 QoS
- LLDP-MED Extended Power-Via-MDI TLV—Power Type, Power Source, Power Priority, Power Value

Optional:

- Port Description
- System Name—Administrator assigned name
- System Description—Includes device type, phone number, hardware version, and software version
- System Capabilities—Set as 'Telephone' capability
- MAC / PHY config status—Detects duplex mismatch
- Management Address—Used for network discovery
- LLDP-MED Location Identification—Location data formats: Co-ordinate, Civic Address, ECS ELIN
- LLDP-MED Inventory Management —Hardware Revision, Firmware Revision, Software Revision, Serial Number, Manufacturer's Name, Model Name, Asset ID

An LLDP frame shall contain all mandatory TLVs. The frame will be recognized as LLDP only if it contains mandatory TLVs. Polycom phones running the UC Software will support LLDP frames with both mandatory and optional TLVs. The basic structure of an LLDP frame and a table containing all TLVs along with each field is explained in [Supported TLVs](#).

LLDP-MED Location Identification

As per section 10.2.4.4 of the LLDP-MED standard, LLDP-MED endpoint devices need to transmit Location Identification TLVs if they are capable of either automatically determining their physical location by use of GPS or radio beacon or capable of being statically configured with this information.

At present, the phones do not have the capability to determine their physical location automatically or provision to a statically configured location. Because of these limitations, the phones will not transmit Location Identification TLV in the LLDP frame. However, the location information from the switch is decoded and displayed on the phone's menu.

For more information on device configuration parameters, refer to [<device/>](#).

Supported TLVs

The basic TLV format is as follows:

- TLV Type (7 bits) [0-6]
- TLV Length (9 bits) [7-15]
- TLV Information (0-511 bytes)

The following is a list of supported TLVs.

Table 12-13: Supported TLVs

| No | Name | Type(7 bits) [0-6] | Length (9 bits) [7-15] | Type Length | Org. Unique Code (3 bytes) | Sub Type |
|----|--|--------------------|-----------------------------|---------------|----------------------------|----------|
| 1 | Chassis-Id¹ | 1 | 6 | 0x0206 | - | 5 |
| | IP address of phone (4 bytes). Note that 0.0.0.0 is not sent until the phone has a valid IP address. | | | | | |
| 2 | Port-Id¹ | 2 | 7 | 0x0407 | - | 3 |
| | MAC address of phone (6 bytes) | | | | | |
| 3 | TTL | 3 | 2 | 0x0602 | - | - |
| | TTL value is 120/0 sec | | | | | |
| 4 | Port description | 4 | 1 | 0x0801 | - | - |
| | Port description 1 | | | | | |
| 5 | System name | 5 | min len > 0, max len <= 255 | - | - | - |

Refer to [System and Model Names](#).

| No | Name | Type(7 bits) [0-6] | Length (9 bits) [7-15] | Type Length | Org. Unique Code (3 bytes) | Sub Type |
|--|---|--------------------|-----------------------------|-------------|----------------------------|----------|
| 6 | System description | 6 | min len > 0, max len <= 255 | - | - | - |
| <p>Manufacturer's name - "Polycom"; Refer to Error! Reference source not found.; Hardware version; Application version; BootROM version</p> | | | | | | |
| 7 | Capabilities | 7 | 4 | 0x0e04 | - | - |
| <p>System Capabilities: Telephone and Bridge if the phone has PC port support and it is not disabled. Enabled Capabilities: Telephone and Bridge if phone has PC port support, it is not disabled and PC port is connected to PC.
Note: PC port supported Phones: IP 330, IP 331, IP 335, IP 450, IP 550, IP 560, and IP 650.</p> | | | | | | |
| 8 | Management Address | 8 | 12 | 0x100c | - | - |
| <p>Address String Len - 5, IPV4 subtype, IP address, Interface subtype - "Unknown", Interface number - "0", ODI string Len - "0"</p> | | | | | | |
| 9 | IEEE 802.3 MAC/PHY config/status¹ | 127 | 9 | 0xfe09 | 0x00120f | 1 |
| <p>Auto Negotiation Supported - "1", enabled/disabled, Refer to PMD Advertise and Operational MAU.</p> | | | | | | |
| 10 | LLDP-MED capabilities | 127 | 7 | 0xfe07 | 0x0012bb | 1 |
| <p>Capabilities - 0x33 (LLDP-Med capabilities, Network policy, Extended Power Via MDI-PD, Inventory) Class Type III
Note: Once support for configuring location Identification information is locally available:
Capabilities - 0x37 (LLDP-Med capabilities, Network policy, Location Identification, Extended Power Via MDI-PD, Inventory) Class Type III</p> | | | | | | |
| 11 | LLDP-MED network policy² | 127 | 8 | 0xfe08 | 0x0012bb | 2 |
| <p>ApplicationType: Voice (1), Policy: (Unknown(=1)/Defined(=0) Unknown, if phone is in booting stage or if switch doesn't support network policy TLV. Defined, if phone is operational stage and Networkpolicy TLV is received from the switch.), Tagged/Untagged, VlanId, L2 priority and DSCP</p> | | | | | | |
| 12 | LLDP-MED network policy² | 127 | 8 | 0xfe08 | 0x0012bb | 2 |
| <p>ApplicationType: Voice Signaling (2), Policy: (Unknown(=1)/Defined(=0) Unknown, if phone is in booting stage or if switch doesn't support network policy TLV. Defined, if phone is operational stage and Networkpolicy TLV is received from the switch.), Tagged/Untagged, VlanId, L2 priority and DSCP.
Note: Voice signaling TLV is sent only if it contains configuration parameters that are different from voice parameters.</p> | | | | | | |

| No | Name | Type(7 bits) [0-6] | Length (9 bits) [7-15] | Type Length | Org. Unique Code (3 bytes) | Sub Type |
|---|---|--------------------|-----------------------------|-------------|----------------------------|----------|
| 13 | LLDP-MED network policy² | 127 | 8 | 0xfe08 | 0x0012bb | 2 |
| <p>ApplicationType: Video Conferencing (6),Policy: (Unknown(=1)/Defined(=0). Unknown, if phone is in booting stage or if switch doesn't support network policy TLV. Defined, if phone is operational stage and Networkpolicy TLV is received from the switch.),Tagged/Untagged, VlanId, L2 priority and DSCP.
 Note: Video Conferencing TLV is sent only from Video capable phones (currently Polycom VVX 1500 only).</p> | | | | | | |
| 14 | LLDP-MED location identification³ | 127 | min len > 0, max len <= 511 | - | 0x0012bb | 3 |
| <p>ELIN data format: 10 digit emergency number configured on the switch. Civic Address: physical address data such as city, street number, and building information.</p> | | | | | | |
| 15 | Extended power via MDI | 127 | 7 | 0xfe07 | 0x0012bb | 4 |
| <p>PowerType -PD device PowerSource-PSE&local Power Priority -Unknown PowerValue - Refer to Power Values</p> | | | | | | |
| 16 | LLDP-MED inventory hardware revision | 127 | min len > 0, max len <= 32 | - | 0x0012bb | 5 |
| <p>Hardware part number and revision</p> | | | | | | |
| 17 | LLDP-MED inventory firmware revision | 127 | min len > 0, max len <= 32 | - | 0x0012bb | 6 |
| <p>BootROM revision</p> | | | | | | |
| 18 | LLDP-MED inventory software revision | 127 | min len > 0, max len <= 32 | - | 0x0012bb | 7 |
| <p>Application (SIP) revision</p> | | | | | | |
| 19 | LLDP-MED inventory serial number | 127 | min len > 0, max len <= 32 | - | 0x0012bb | 8 |
| <p>MAC Address (ASCII string)</p> | | | | | | |
| 20 | LLDP-MED inventory manufacturer name | 127 | 11 | 0xfe0b | 0x0012bb | 9 |
| <p>Polycom</p> | | | | | | |

| No | Name | Type(7 bits) [0-6] | Length (9 bits) [7-15] | Type Length | Org. Unique Code (3 bytes) | Sub Type |
|--|-------------------------------|--------------------|----------------------------|-------------|----------------------------|----------|
| 21 | LLDP-MED inventory model name | 127 | min len > 0, max len <= 32 | - | 0x0012bb | 10 |
| Refer to Error! Reference source not found. | | | | | | |
| 22 | LLDP-MED inventory asset ID | 127 | 4 | 0xfe08 | 0x0012bb | 11 |
| Empty (Zero length string) | | | | | | |
| 23 | End of LLDP DU | 0 | 0 | 0x0000 | - | - |

¹ For other subtypes, refer to [IEEE 802.1AB](#), March 2005.

² For other application types, refer to [TIA Standards 1057](#), April 2006.

³ At this time, this TLV is not sent by the phone.

System and Model Names

The following table outlines the Polycom phone models, and their system and model names:

Table 12-14: Phone System and Model Names

| Model | System Name | Model Name |
|---------|------------------------------|--------------------------|
| IP 321 | Polycom SoundPoint IP 321 | SoundPointIP-SPIP_321 |
| IP 331 | Polycom SoundPoint IP 331 | SoundPointIP-SPIP_331 |
| IP 335 | Polycom SoundPoint IP 335 | SoundPointIP-SPIP_335 |
| IP 450 | Polycom SoundPoint IP 450 | SoundPointIP-SPIP_450 |
| IP 550 | Polycom SoundPoint IP 550 | SoundPointIP-SPIP_550 |
| IP 560 | Polycom SoundPoint IP 560 | SoundPointIP-SPIP_560 |
| IP 650 | Polycom SoundPoint IP 650 | SoundPointIP-SPIP_650 |
| IP 5000 | Polycom SoundStation IP 5000 | SoundStationIP-SSIP_5000 |
| IP 6000 | Polycom SoundStation IP 6000 | SoundStationIP-SSIP_6000 |
| Duo | Polycom SoundStation Duo | SoundStation-SS_Duo |
| VVX 500 | Polycom VVX 500 | VVX-VVX_500 |

| <i>Model</i> | <i>System Name</i> | <i>Model Name</i> |
|--------------|--------------------|-------------------|
| VVX 1500 | Polycom VVX 1500 | VVX-VVX_1500 |

PMD Advertise and Operational MAU

The following table lists values for the PMD Advertise and Operational MAU.

Table 12-15: PMD Advertise and Operation MAU Type

| <i>Mode/Speed</i> | <i>PMD Advertise Capability Bit</i> | <i>Operational MAU Type</i> |
|-----------------------------|-------------------------------------|-----------------------------|
| 10BASE-T half duplex mode | 1 | 10 |
| 10BASE-T full duplex mode | 2 | 11 |
| 100BASE-T half duplex mode | 4 | 15 |
| 100BASE-T full duplex mode | 5 | 16 |
| 1000BASE-T half duplex mode | 14 | 29 |
| 1000BASE-T full duplex mode | 15 | 30 |
| Unknown | 0 | 0 |



Note: Default PMD Advertise Capability Values

By default, all phones have the PMD Advertise Capability set for 10HD, 10FD, 100HD, and 100FD bits. SoundPoint IP 560 and Polycom VVX 1500 phones that have Gigabit Ethernet support PMD Advertise Capability also contains set 1000FD bit.

Power Values

The following table outlines the power usage for each phone, as well as the power value sent in LLDP-MED.

Table 12-16: Phone Power Values

| <i>Model</i> | <i>Power Usage (Watts)</i> | <i>Power Value Sent in LLDP-MED Extended Power Via MDI TLV</i> |
|--------------|----------------------------|--|
| IP 321 | 3.5 | 35 |
| IP 331 | 3.7 | 37 |
| IP 335 | 3.9 | 39 |

| <i>Model</i> | <i>Power Usage (Watts)</i> | <i>Power Value Sent in LLDP-MED Extended Power Via MDI TLV</i> |
|----------------|----------------------------|--|
| IP 450 | 5.4 | 54 |
| IP 550 | 5.9 | 59 |
| IP 560 | 8.3 | 83 |
| IP 650 with EM | 12 | 120 |
| IP 5000 | 5.8 | 58 |
| IP 6000 | 9.8 | 98 |
| Duo | 7.0 | 70 |
| VVX 500 | 8.0 | 80 |
| VVX 1500 | 11.8 | 118 |



Note: Default Power Values

By default, the power values for the SoundPoint IP 650 are sent for the phone and the Expansion Module(s). The values are not adjusted when the Expansion Module(s) are detached from the phone.

Part V: Polycom[®] UC Software Reference Information

Part V provides you with detailed information about the configuration files you need to download to your provisioning server to deploy your Polycom[®] phones and to configure basic, advanced, audio, video, and user and phone security features. Part VI also provides reference information about the Session Initiation Protocol (SIP), the Polycom UC Software menu structure as it displays on most Polycom phones, and information about the third-party software that is included in the Polycom UC Software.

Part VI consists of the following chapters:

- [Chapter 13: Configuration Parameters](#)
- [Chapter 14: Session Initiation Protocol \(SIP\)](#)
- [Chapter 15: Polycom UC Software Menu System](#)
- [Chapter 16: Third Party Software](#)

Chapter 13: Configuration Parameters

This chapter is a reference guide to the UC Software configuration parameters used to configure all phone features and functions. This chapter is useful if you want to read a detailed description of a particular configuration parameter or you would like to see the default or permitted values for that parameter. If you would like to configure a specific feature, you should find the feature in [Part III: Configuring Your System](#). These parameters in this section include:

- [<acd/>](#)
- [<apps/>](#)
- [<attendant>](#)
- [<bg/>](#)
- [<bitmap/>](#)
- [<bluetooth/>](#)
- [<button/>](#)
- [<call/>](#)
- [<callLists/>](#)
- [<device/>](#)
- [<dialplan/>](#)
- [<dir>](#)
 - [<local/>](#)
 - [<corp/>](#)
- [<divert/>](#)
- [<dns/>](#)
 - [DNS-A](#)
 - [DNS-NAPTR](#)
 - [DNS-SRV](#)
- [<efk/>](#)
- [<exchange/>](#)
- [<feature/>](#)
- [](#)
- [<httpd/>](#)
- [<key/>](#)
- [<keypadLock/>](#)

- <lcl/>
 - <ml/>
 - <datetime/>
- <license/>
- <lineKey/>
- <log/>
 - <level/> <change/>and<render/>
 - <sched/>
- <mb/>
- <messaging/>
- <msg/>
- <nat/>
- <np/>
- <oai/>
- <phoneLock/>
- <pnet/>
- <powerSaving/>
- <pres/>
- <prov/>
- <ptt/>
- <qbc/>
- <qos/>
- <reg/>
- <request/>
- <roaming_buddies/>
- <roaming_privacy/>
- <saf/>
- <se/>
 - <pat/>
 - <rt/>
- <sec/>
 - <encryption/>
 - <pwd/><length/>
 - <srtp/>

-
- <H235/>
 - <dot1x><eapollogoff/>
 - <hostmovedetect/>
 - <TLS/>
 - » <profile/>
 - » <profileSelection/>
 - <softkey/>
 - <tcpIpApp/>
 - <dns/>
 - <sntp/>
 - <port/><rtp/>
 - <keepalive/>
 - <tones/>
 - <DTMF/>
 - <chord/>
 - <up/>
 - <upgrade/>
 - <video/>
 - <codecs/>
 - » <codecPref/>
 - <profile/>
 - <camera/>
 - <localCameraView/>
 - <voice/>
 - <codecPref/>
 - <volume/>
 - <vad/>
 - <quality monitoring/>
 - <rxQoS/>
 - <volpProt/>
 - <server/>
 - <SDP/>
 - <SIP/>
 - <H323/>

- `<webutility/>`
- `<Wi-Fi/>`

`<acd/>`

The SIP-B Automatic Call Distribution (ACD) and Feature Synchronized ACD features use the `<acd/>` parameter, shown in the following table.

Table 13-1: Automatic Call Distribution Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| acd.reg¹ | 1 to 34 | 1 |
| The index of the registration (line) used to support BroadSoft server-based ACD. | | |
| acd.stateAtSignIn | 0 or 1 | 1 |
| The state of the user when signing in. If 1, the user is available. If 0, the user is unavailable. | | |
| acd.x.unavailreason.active
If 1, the reason code is active. If 0, the code is inactive. | 0 or 1 | 0 |
| acd.x.unavailreason.codeValue¹
The code value. For example, <i>1000100000</i> | String | Null |
| acd.x.unavailreason.codeName¹
The code name. For example, <i>Out to Lunch</i> | string | Null |
| These three parameters configure the unavailable reason codes used for premium feature-synchronized ACD features, where x is the index of up to 100 codes. | | |

¹ Change causes phone to restart or reboot.

`<apps/>`

The `<apps/>` parameter is used to control telephone notification events, state polling events, and push server controls. For more information, see the [Polycom Web Application Developer's Guide](#).

Table 13-2: Application Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| apps.push.alertSound | 0 or 1 | 0 |
| If 0, there is no sound when an alert is pushed. If 1, there is sound. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| apps.push.messageType | 0 to 5 | 0 |
| Choose a priority level for push messages from the application server to the phone.
0: (None) 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 | string | null |
| The password to access the push server URL. | | |
| apps.push.secureTunnelEnabled | 0 or 1 | 1 |
| If 0, the Web server is not connected through a secure tunnel. If 1, the Web server is connected through a secure tunnel. | | |
| apps.push.secureTunnelPort | 1 to 65535 | 443 |
| The port that the phone should use to communicate to the Web server when the secure tunnel is used. | | |
| apps.push.secureTunnelRequired | 0 or 1 | 0 |
| If 0, communications to the Web server do not require a secure tunnel. If 1, communications require a secure tunnel. | | |
| apps.push.serverRootURL | URL | null |
| 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 <code>http://172.24.128.85:8080/sampleapps</code> and the relative URL is <code>/examples/sample.html</code> , the URL that is sent to the microbrowser is <code>http://172.24.128.85:8080/sampleapps/examples/sample.html</code> . Can be either HTTP or HTTPS. | | |
| apps.push.username | string | null |
| The user name to access the push server URL.
<i>Note:</i> To enable the push functionality, the parameters <code>apps.push.username</code> and <code>apps.push.password</code> must be set (not null). | | |
| apps.statePolling.password | string | null |
| Enter the password that the phone requires to authenticate phone state polling. | | |
| apps.statePolling.URL | URL | null |
| The URL to which the phone sends call processing state/device/network information. The protocol used can be either HTTP or HTTPS. <i>Note:</i> To enable state polling, the parameters <code>apps.statePolling.URL</code> , <code>apps.statePolling.username</code> , and <code>apps.statePolling.password</code> must be set to non-null values. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| apps.statePoling.responseMode | 0 or 1 | 1 |
| The mode of sending requested polled data. If 1, requested polled data is sent to a configured URL. If 0, the data is sent in the HTTP response. | | |
| apps.statePolling.username | string | null |
| Enter the user name that the phone requires to authenticate phone state polling. | | |
| apps.telNotification.callStateChangeEvent | 0 or 1 | 0 |
| If 0, call state change notification is disabled. If 1, notification is enabled. | | |
| apps.telNotification.incomingEvent | 0 or 1 | 0 |
| If 0, incoming call notification is disabled. If 1, notification is enabled. | | |
| apps.telNotification.lineRegistrationEvent | 0 or 1 | 0 |
| If 0, line registration notification is disabled. If 1, notification is enabled. | | |
| apps.telNotification.networkUpEvent | 0 or 1 | 0 |
| If 0, network up notification is disabled. If 1, notification is enabled. | | |
| apps.telNotification.offhookEvent | 0 or 1 | 0 |
| If 0, off-hook notification is disabled. If 1, notification is enabled. | | |
| apps.telNotification.onhookEvent | 0 or 1 | 0 |
| If 0, on-hook notification is disabled. If 1, notification is enabled. | | |
| apps.telNotification.outgoingEvent | 0 or 1 | 0 |
| If 0, outgoing call notification is disabled. If 1, notification is enabled. | | |
| apps.telNotification.uiInitializationEvent | 0 or 1 | 0 |
| If 0, user interface initialization notification is disabled. If 1, notification is enabled. | | |
| apps.telNotification.URL | URL | null |
| The URL to which the phone sends notifications of specified events. Can be either HTTP or HTTPS. | | |
| apps.telNotification.x.URL | URL | null |
| The URL to which the phone sends notifications of specified events, where x 1 to 9. Can be either HTTP or HTTPS. | | |
| apps.telNotification.userLoginOutEvent | 0 or 1 | 0 |
| If 0, user login/logout notification is disabled. If 1, notification is enabled. | | |
| apps.ucdesktop.adminEnabled¹ | 0 or 1 | 1 |
| If 0, the Polycom Desktop Connector is disabled on the administrative level. If 1, it is enabled on the administrative level. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---------------------------------|--------------------|
| apps.ucdesktop.desktopUserName | string | null |
| The user's name, supplied from the user's computer. For example, <i>bsmith</i> . | | |
| apps.ucdesktop.enabled | 0 or 1 | 0 |
| If 0, the Polycom Desktop Connector is disabled for users. If 1, it is enabled for users. | | |
| apps.ucdesktop.orientation | Unspecified, Left, Right | Unspecified |
| The location of the VVX 500 and 1500 with respect to the user's computer. For example, to the <code>Left</code> of the computer. | | |
| apps.ucdesktop.ServerAddress | string | null |
| The user's computer as a fully qualified domain name (FQDN). For example, <i>computer@yourcompany.com</i> . | | |
| apps.ucdesktop.ServerPort | 1 to 65535 | 24800 |
| The port number. <i>Note:</i> This value should be the same as the one that is used on the user's computer, otherwise the connection is not established. | | |
| apps.x.label²
The descriptive text that displays in the Applications menu | String | null |
| apps.x.url²
The URL of an application | | |
| The label and URL of up to 12 applications (for x = 1 to 12). | | |

¹ Change causes phone to restart or reboot.

² This parameter is supported on only the SpectraLink 8400 Series handsets.

<attendant>

These parameters are only supported on the SoundPoint IP 450, 550, 560, and 650 phones.

The Busy Lamp Field (BLF)/attendant console feature enhances support for phone-based monitoring.

In the following table, x is the monitored user number. For IP 450: x=1-2; IP 550, IP 560: X=1-3; IP 650.

Table 13-3: Attendant/Busy Lamp Field Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------------------|----------------|
| attendant.reg¹ | positive integer | 1 |
| The index of the registration that will be used to send a SUBSCRIBE to the list SIP URI specified in <code>attendant.uri</code> . For example, <code>attendant.reg = 2</code> means the second registration will be used. | | |
| attendant.ringType | default, ringer1 to ringer24 | ringer1 |
| The ringtone to play when a BLF dialog is in the offering state. | | |
| attendant.uri¹ | string | Null |
| The list SIP URI on the server. If this is just a user part, the URI is constructed with the server hostname/IP. | | |
| <i>Note:</i> If this parameter is set, then the individually addressed users configured by <code>attendant.resourceList</code> and <code>attendant.behaviors</code> are ignored | | |
| attendant.behaviors.display.spontaneousCallAppearances.normal¹ | 0 or 1 | 1 |
| Normal | 0 or 1 | 0 |
| attendant.behaviors.display.spontaneousCallAppearances.automata¹ | | |
| Automatic | | |
| If 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). If 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. | | |
| attendant.behaviors.display.remoteCallerID.normal¹ | 0 or 1 | 1 |
| Normal | | |
| attendant.behaviors.display.remoteCallerID.automata¹ | | |
| Automatic | | |
| If 1, normal and automatic remote party caller ID information is presented to the attendant. If 0, the string <code>unknown</code> will be substituted for both name and number information. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---|----------------|
| attendant.resourceList.x.address | string that constitutes a valid SIP URI (sip:6416@polycor.com) or contains the user part of a SIP URI (6416) | Null |
| <p>The user referenced by <code>attendant.reg=""</code> will subscribe to this URI for dialog. If a user part is present, the phone will subscribe to a sip URI constructed from the user part and domain of the user referenced by <code>attendant.reg</code>.</p> | | |
| attendant.resourceList.x.callAddress¹ | string | Null |
| <p>If the BLF call server is not at the same address as the BLF presence server, calls will be sent to this address instead of the address specified by <code>attendant.resourceList.x.address</code>.</p> | | |
| attendant.resourceList.x.label | UTF-8 encoded string | Null |
| <p>The text label displays adjacent to the associated line key. If set to Null, the label will be derived from the user part of <code>attendant.resourceList.x.address</code>.</p> | | |
| attendant.resourceList.x.proceedingsRecipient¹ | 0 or 1 | 0 |
| <p>A flag to determine if pressing the associated line key for the monitored user will pick up the call.</p> | | |
| attendant.resourceList.x.type | normal or automata | normal |
| <p>The type of resource being monitored and the default action to perform when pressing the line key adjacent to monitored user <i>x</i>.
 If <code>normal</code>, 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.
 If <code>automata</code>, the default action when 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</p> | | |

¹ Change causes phone to restart or reboot.

<bg/>

This section defines the backgrounds you can display on the SoundPoint IP 450, 550, 560, and 650, and the VVX 500 and 1500 phones. SoundPoint IP 550, 560, and 650 phones use `hiRes` parameters. SoundPoint IP 450 phones use `medRes` parameters. SoundPoint IP 550, 560, and 650 phones use `hiRes.gray` parameters. VVX 500 and 1500 phones use `VVX_1500` parameters.

Table 13-4: Background Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|----------------------------------|
| bg.color.selection | w,x | 1,1 |
| <p>Set the background. Specify which type of background (w) and index (x) for that type is selected on reboot. The default selection is 2,1 the first solid background.</p> <p>Use w=1 and x=1 (1,1) to select the built-in image.</p> <p>Use w=2 and x= 1 to 4 to select one of the four <code>solid</code> backgrounds.</p> <p>Use w=3 and x= 1 to 6 to select one of the six background <code>bm</code> images</p> | | |
| bg.color.bm.x.name
Phone screen background image file | URL or file path of a BMP or JPEG image | built-in value of Thistle |
| bg.color.bm.x.em.name
Expansion module (EM) background image file | URL or file path of a BMP or JPEG image | |
| <p>The name of the image file (including extension). The six (x: 1 to 6) default screen and expansion module (EM) background images are:</p> <ul style="list-style-type: none"> x=1: Leaf.jpg and LeafEM.jpg x=2: Sailboat.jpg and SailboatEM.jpg x=3: Beach.jpg and BeachEM.jpg x=4: Palm.jpg and PalmEM.jpg x=5: Jellyfish.jpg and JellyfishEM.jpg x=6: Mountain.jpg and MountainEM.jpg <p><i>Note:</i> If the file is missing or unavailable, the built-in default solid pattern is displayed.</p> | | |
| bg.VVX_1500.color.selection | w,x | 1,1 |
| <p>Set the background for the VVX 500 and 1500. Specify which type of background (w) and index for that type (x) is selected on reboot where w=1 to 3, x=1 to 6. The default selection is 1,1 – the built-in background.</p> <p>w=1 is used with x=1 to select the built-in background (use 1,1).</p> <p>w=2 is used when selecting an image as a background.</p> <p>w=3 is used with x=1 to select the <i>Local File</i> Digital Picture Frame image as a background (use 3,1). Only one local file at a time is supported.</p> | | |

| Parameter | Permitted Values | Default |
|---|--|-------------|
| bg.VVX_1500.color.bm.x.name | URL or file path of a JPEG, BMP, or PNG image | Null |
| <p>The name of the image file (including the extension). For example, <i>PolycomLogo.bmp</i> will load the PolycomLogo BMP file from the provisioning server while <i>http://mysite.com/myLogo.png</i> will load MyLogo from <i>mysite</i>. Images will be available in the phone's background menu.</p> | | |
| bg.hiRes.gray.selection | w,x | 2,1 |
| <p>Set the background on the SoundPoint IP 550, 560, or 650. Specify which type of background (w) and index (x) for that type is selected on reboot. The default selection is 2,1 the first solid background.</p> <p>Use w=1 and x=1 (1,1) to select the built-in image.</p> <p>Use w=2 and x= 1 to 4 to select one of the four <code>solid</code> backgrounds.</p> <p>Use w=3 and x= 1 to 6 to select one of the six background <code>bm</code> images.</p> | | |
| bg.hiRes.gray.pat.solid.x.name | any string | |
| Solid pattern name | | |
| bg.hiRes.gray.pat.solid.x.red | 9 to 255 | |
| bg.hiRes.gray.pat.solid.x.green | 9 to 250 | |
| bg.hiRes.gray.pat.solid.x.blue | 9 to 255 | |
| 9,9,9: Dark Gray. 255, 255, 255: White | All three must be the same | |
| <p>Specify up to four (x: 1 to 4) solid grayscale backgrounds for the SoundPoint IP 550, 560, or 650. The defaults are:</p> <p>x=1: White red (255), green (255), blue (255)</p> <p>x=2: Light Gray red (160), green (160), blue (160)</p> <p>x=3 and x=4 Null</p> <p><i>Note:</i> The red, green, and blue values must be the same for the pattern to display correctly.</p> | | |
| bg.hiRes.gray.bm.x.name | URL or file path of a BMP or JPEG image | |
| Phone screen background image file | | |
| bg.hiRes.gray.bm.x.em.name | URL or file path of a BMP or JPEG image | |
| Expansion module (EM) background image file | | |
| bg.hiRes.gray.bm.x.adj | -8 to 3 | |
| Brightness adjustment. -8: Lighter 3: Darker | | |
| <p>The name of the image file (including extension). The six (x: 1 to 6) default screen and expansion module (EM) background images with their brightness adjustments are:</p> <p>x=1: Leaf.jpg and LeafEM.jpg, adjustment: 0</p> <p>x=2: Sailboat.jpg and SailboatEM.jpg, adjustment: -3</p> <p>x=3: Beach.jpg and BeachEM.jpg, adjustment: 0</p> <p>x=4: Palm.jpg and PalmEM.jpg, adjustment: -2</p> <p>x=5: Jellyfish.jpg and JellyfishEM.jpg, adjustment: -2</p> <p>x=6: Mountain.jpg and MountainEM.jpg, adjustment: 0</p> <p><i>Note:</i> If the file is missing or unavailable, the built-in default solid pattern is displayed. The adjustment values are changed on each individual phone when the user lightens or darkens the image during preview</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|----------------|
| bg.medRes.gray.selection | w,x | 2,1 |
| <p>Set the background on the SoundPoint IP 450. Specify which type of background (w) and index (x) for that type is selected on reboot. The default selection is 2,1 the first solid background.</p> <p>Use w=1 and x=1 (1,1) to select the built-in image.</p> <p>Use w=2 and x= 1 to 4 to select one of the four <code>solid</code> backgrounds.</p> <p>Use w=3 and x= 1 to 6 to select one of the six background <code>bm</code> images.</p> | | |
| bg.medRes.gray.pr.x.adj | -8 to 3 | -3 |
| <p>This parameter is not used.</p> | | |
| bg.medRes.gray.pat.solid.x.name | any string | |
| <p>Solid pattern name</p> | | |
| bg.medRes.gray.pat.solid.x.red | 9 to 255 | |
| bg.medRes.gray.pat.solid.x.green | 9 to 255 | |
| bg.medRes.gray.pat.solid.x.blue | 9 to 255 | |
| <p>If red,green,blue = 9: Dark Gray. If 255, White. All three must be the same</p> | | |
| <p>Specify up to four (x: 1 to 4) solid grayscale backgrounds for the SoundPoint IP 550, 560, or 650. The defaults are:</p> <p>x=1: White red (255), green (255), blue (255)</p> <p>x=2: Light Gray red (160), green (160), blue (160)</p> <p>x=3 and x=4 Null</p> <p><i>Note:</i> The red, green, and blue values must be the same for the pattern to display correctly.</p> | | |
| bg.medRes.gray.bm.x.name | URL or file path of a BMP or JPEG image | |
| <p>Phone screen background image file</p> | | |
| bg.medRes.gray.bm.x.adj | -8 to 3 | |
| <p>Brightness adjustment. -8: Lighter 3: Darker</p> | | |
| <p>The name of the image file (including extension). The six (x: 1 to 6) default images and brightness adjustments are:</p> <p>x=1: Leaf256x116.jpg, adjustment: 0</p> <p>x=2: Sailboat256x116.jpg, adjustment: -3</p> <p>x=3: Beach256x116.jpg, adjustment: 0</p> <p>x=4: Palm256x116.jpg, adjustment: -2</p> <p>x=5: Jellyfish256x116.jpg, adjustment: -2</p> <p>x=6: Mountain256x116.jpg, adjustment: 0</p> <p><i>Note:</i> If the file is missing or unavailable, the built-in default solid pattern is displayed. The adjustment values are changed on each individual phone when the user lightens or darkens the image during preview</p> | | |

<bitmap/>

You can display a custom image on the idle display of SoundPoint IP and SoundStation IP phones. Soft keys and line keys do not block or cover idle display bitmaps. You can create a custom image such as a company logo to display on the idle screen at all times.

Table 13-5: Idle Display Bitmap Parameters

| <i>Parameter</i> | <i>Permitted Values</i> |
|--|--|
| bitmap.idleDisplay.name¹ | URL or file path of a BMP or JPEG image |
| The name of the image file (including the extension). For example, <i>PolycomLogo.bmp</i> will load the PolycomLogo BMP file from the provisioning server while <i>http://mysite.com/myLogo.jpeg</i> will load MyLogo from <i>mysite</i> . | |

¹ Change causes phone to restart or reboot.

<bluetooth/>

Bluetooth headsets are supported with only the SpectraLink handsets.

Table 13-6: Bluetooth Radio Transmitter Parameter

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| bluetooth.radioOn | 0 or 1 | 0 |
| If 0, the Bluetooth radio (transmitter/receiver) is off. If 1, the Bluetooth radio is on. The Bluetooth radio must be turned on before the phone can use a Bluetooth headset. | | |

<button/>

You can configure the color of line keys and soft keys using the <button/> parameter. This parameter is not supported on the VVX 500 and 1500 phones or the SpectraLink handsets.

Table 13-7: Soft Key Button Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| button.color.selection.x.y.modify | any string | |
| <p>The label color for soft keys and line key labels associated with the defined colored backgrounds. These values can be modified locally by the user.</p> <p>The format is: <code>rgbHILO, <parameter list></code>. For example: <code>rbgHiLo, 51, 255, 68, 255, 0, 119</code> is the default button color associated with the built-in background.</p> | | |
| button.gray.selection.x.y.modify | any string | |
| <p>The label color for soft keys and line key labels associated with the defined gray backgrounds. These values can be modified locally by the user.</p> <p>The format is: <code>rgbHILO, <parameter list></code>. By default, all defaults are set to <code>none</code>.</p> | | |

<call/>

The phone supports an optional per-registration feature that enables automatic call placement when the phone is off-hook.

The phone supports a per-registration configuration that determines which events will cause the missed-calls counter to increment.

You can enable/disable missed call tracking on a per-line basis.



Note: Reading the Call Parameter Table

In the following table, x is the registration number. IP 321/331/335: x=1-2; IP 450: x=1-3; IP 550, 560: x=1-4; VVX 500:x=1-12; VVX 1500: x=1-6; IP 650, IP 5000, 6000, 7000: x=1.

This per-site and per-phone configuration parameters are defined as follows:

Table 13-8: Call Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| call.advancedMissedCalls.addToReceivedList | 0 or 1 | 0 |
| <p>Applies to calls on shared lines that are answered remotely. If 0, such calls are not added to the local receive call list. If 1, the calls are added to the local receive call list.</p> | | |
| call.advancedMissedCalls.enabled | 0 or 1 | 1 |
| <p>If 1, improved missed call handling for shared lines is enabled (shared lines can correctly count missed calls). If 0, the old missed call handling is used for shared lines (shared lines may not correctly count missed calls).</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|-----------------------|
| call.advancedMissedCalls.reasonCodes | comma-separated list of indexes | 200 |
| A comma separated list of reason code indexes that are interpreted to mean that a call should not be considered as a missed call. | | |
| call.autoAnswer.H323 | 0 or 1 | 0 |
| VVX 1500 only. If 0, auto-answer is disabled for H.323 calls. If 1, auto-answer is enabled for all H.323 calls. | | |
| call.autoAnswer.micMute | 0 or 1 | 1 |
| If 0, the microphone is active immediately after a call is auto-answered. If 1, the microphone is initially muted after a call is auto-answered. | | |
| call.autoAnswer.ringClass | see the list of ring classes in <rt/> | ringAutoAnswer |
| The ring class to use when a call is to be automatically answered using the auto-answer feature. If set to a ring class with a type other than <code>answer</code> or <code>ring-answer</code> , the setting will be overridden such that a ringtone of <code>visual</code> (no ringer) applies. | | |
| call.autoAnswer.SIP | 0 or 1 | 0 |
| VVX 1500 only. If 0, auto-answer is disabled for SIP calls. If 1, auto-answer is enabled for all SIP calls. | | |
| call.autoAnswer.videoMute | 0 or 1 | 0 |
| VVX 1500 only. If 0, video begins transmitting (video Tx) immediately after a call is auto-answered. If 1, video transmission (video Tx) is initially disabled after a call is auto-answered. | | |
| call.autoOffHook.x.enabled¹ | 0 or 1 | 0 |
| Enable or disable the feature | | |
| call.autoOffHook.x.contact¹ | a SIP URL | Null |
| The contact address to where the call is placed | | |
| call.autoOffHook.x.protocol¹ | SIP or H323 | Null |
| The calling protocol to use | | |
| If <code>enabled</code> is set to 0, no call is placed automatically when the phone goes off hook, and the other parameters are ignored. If <code>enabled</code> is set to 1, a call is automatically placed to the <code>contact</code> using the calling <code>protocol</code> , when the phone goes off hook. | | |
| Only the VVX 1500 phone uses the <code>protocol</code> parameter, if no protocol is specified, the VVX 1500 phone will use the protocol specified by <code>call.autoRouting.preferredProtocol</code> . If a line is configured for a single protocol, the configured protocol will be used. | | |
| The <code>contact</code> must be an ASCII-encoded string containing digits, either the user part of a SIP URL (for example, <code>6416</code>), or a full SIP URL (for example, <code>6416@polycom.com</code>). | | |
| call.autoRouting.preferredProtocol | SIP or H323 | SIP |
| VVX 1500 only. If set to SIP , calls are placed via SIP if available, or via H.323 if SIP is not available. If set to H323 , calls are placed via H.323 if available, or via SIP if H.323 is not available. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---------------------------|-----------------|
| call.autoRouting.preference | line or protocol | line |
| <p>VVX 1500 only. If set to line, calls are placed via the first available line, regardless of its protocol capabilities. If the first available line has both SIP and H.323 capabilities, the preferred protocol will be used (<code>call.autoRouting.preferredProtocol</code>).</p> <p>If set to protocol, the first available line with the preferred protocol activated is used, if available. If not available, the first available line will be used.</p> <p>Note: Auto-routing is used when manual routing selection features (<code>up.manualProtocolRouting</code>) are disabled.</p> | | |
| call.callsPerLineKey | 1-4, 1-8, 1-24 | 4, 8, 24 |
| <p>Set the maximum number of concurrent calls per line key. This parameter applies to all registered lines. For the SoundPoint IP 321/331/335, the permitted range is 1 to 4 and the default is 4. For the SoundPoint IP 550, 560, and 650 the permitted range is 1 to 24 and the default is 24. For all other phones, the permitted range is 1 to 8 and the default is 8.</p> <p>Note that this parameter may be overridden by the per-registration parameter of <code>reg.x.callsPerLineKey</code>.</p> | | |
| call.callWaiting.enable | 0 or 1 | 1 |
| <p>If 1, the phone alerts you to an incoming call while you are in an active call. If 0, you are not alerted to incoming calls while in an active call and the incoming call is treated as if you did not answer it. If 1, and you end the active call during a second incoming call, you are alerted to the second incoming call.</p> | | |
| call.callWaiting.ring¹ | beep, ring, silent | beep |
| <p>Specifies the ringtone of incoming calls when another call is active. If set to Null, the default value is beep.</p> | | |
| call.dialtoneTimeOut¹ | positive integer | 60 |
| <p>The time in seconds that a dial tone will play before a call is dropped. If set to 0, the call is not dropped.</p> | | |
| call.directedCallPickupMethod¹ | native or legacy | Null |
| <p>Specifies how the phone will perform a directed call pick-up from a BLF contact. native indicates the phone will use a native protocol method (in this case SIP INVITE with the Replaces header). legacy indicates the phone will use the method specified in <code>call.directedCallPickupString</code>.</p> | | |
| call.directedCallPickupString¹ | star code | *97 |
| <p>The star code to initiate a directed call pickup. <i>Note:</i> The default value supports the BroadWorks calls server only. You must change the value if your organization uses a different call server.</p> | | |
| call.donotdisturb.perReg¹ | 0 or 1 | 0 |
| <p>This parameter determines if the Do-Not-Disturb feature will apply to all registrations on the phone (globally), or apply on a per-registration basis. If 0, DND will apply to all registrations on the phone when it is active. If 1, the user can activate DND on a per-registration basis. <i>Note:</i> If <code>voIpProt.SIP.serverFeatureControl.dnd</code> is set to 1 (enabled), this parameter is ignored.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|------------------------------------|----------------|
| call.enableOnNotRegistered¹ | 0 or 1 | 1 |
| If 1, users can make calls when the phone is not registered. If 0, calls are not permitted without registration. <i>Note:</i> Setting this parameter to 1 can allow Polycom VVX 1500 phones to make calls using the H.323 protocol even though an H.323 gatekeeper is not configured. | | |
| call.hold.localReminder.enabled¹ | 0 or 1 | 0 |
| If 1, users are reminded of calls that have been on hold for an extended period of time. If 0, there is no hold reminder. | | |
| call.hold.localReminder.period¹ | non-negative integer | 60 |
| Specify the time in seconds between subsequent hold reminders. | | |
| call.hold.localReminder.startDelay¹ | non-negative integer | 90 |
| Specify a time in seconds to wait before the initial hold reminder. | | |
| call.lastCallReturnString¹ | string of maximum length 32 | *69 |
| The string sent to the server when the user selects the last call return action. The string is usually a star code. | | |
| call.localConferenceCallHold¹ | 0 or 1 | 0 |
| If set to 0, a hold will happen for all legs when conference is put on hold. (old behavior).
If set to 1, only the host is out of the conference, all other parties in conference continue to talk. (new behavior).
Only supported for the SoundPoint IP 550, 560, and 650. For all others, set to 0. | | |
| call.localConferenceEnabled¹ | 0 or 1 | 0 |
| If set to 0, the Conference and Join soft keys do not display during an active call and you cannot establish conferences on the phone.
If set to 1, the Conference and Join soft keys display during an active call and you can establish conferences on the phone. | | |
| call.missedCallTracking.x.enabled¹ | 0 or 1 | 1 |
| If set to 1, missed call tracking is enabled.
If <code>call.missedCallTracking.x.enabled</code> is set to 0, then missedCall counter is not updated regardless of what <code>call.serverMissedCalls.x.enabled</code> is set to (and regardless of how the server is configured). There is no Missed Call List provided under Menu > Features of the phone.
If <code>call.missedCallTracking.x.enabled</code> is set to 1 and <code>call.serverMissedCalls.x.enabled</code> is set to 0, then the number of missedCall counter is incremented regardless of how the server is configured.
If <code>call.missedCallTracking.x.enabled</code> is set to 1 and <code>call.serverMissedCalls.x.enabled</code> is set to 1, then the handling of missedCalls depends on how the server is configured. | | |
| call.offeringTimeOut¹ | positive integer | 60 |
| Specify a time in seconds that an incoming call will ring before the call is dropped, 0=infinite.
<i>Note:</i> The call diversion, no answer feature will take precedence over this feature if enabled. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| call.parkedCallRetrieveMethod¹ | native or legacy | Null |
| The method the phone will use to retrieve a BLF resource's call which has dialog state confirmed. native indicates the phone will use a native protocol method (in this case SIP INVITE with the Replaces header). legacy indicates the phone will use the method specified in <code>call.parkedCallRetrieveString</code> . | | |
| call.parkedCallRetrieveString¹ | star code | Null |
| The star code used to initiate retrieval of a parked call. | | |
| call.rejectBusyOnDnd¹ | 0 or 1 | 1 |
| If 1, and DND is turned on, the phone rejects incoming calls with a busy signal. If set to 0, and DND is turned on, the phone gives a visual alert of incoming calls and no audio ringtone alert.
<i>Note:</i> This parameter does not apply to shared lines since not all users may want DND enabled. | | |
| call.ringBackTimeOut¹ | positive integer | 60 |
| Specify a time in seconds to allow an outgoing call to remain in the ringback state before dropping the call, 0=infinite. | | |
| call.serverMissedCall.x.enabled¹ | 0 or 1 | 0 |
| If 0, all missed-call events will increment the counter. If set to 1, only missed-call events sent by the server will increment the counter. <i>Note:</i> This feature is supported with the BroadSoft® Synergy call server only (previously known as Sylantro). | | |
| call.shared.disableDivert¹ | 0 or 1 | 1 |
| If set to 1, the diversion feature for shared lines is disabled. <i>Note:</i> This feature is disabled on most call servers. | | |
| call.shared.exposeAutoHolds¹ | 0 or 1 | 0 |
| If 1, a re-INVITE will be sent to the server when setting up a conference on a shared line. If 0, no re-INVITE will be sent to the server. | | |
| call.shared.oneTouchResume¹ | 0 or 1 | 0 |
| If set to 1, all users on a shared line can resume held calls by pressing the shared line key. If more than one call is on hold, the first held call is selected and resumed.
If set to 0, selecting the shared line opens all current calls that the user can choose from.
<i>Note:</i> This parameter applies to the SoundStation IP phones. For other phones, a quick press and release of the line key will resume a call whereas pressing and holding down the line key will show a list of calls on that line. | | |
| call.shared.seizeFailReorder¹ | 0 or 1 | 1 |
| If set to 1, play re-order tone locally on shared line seize failure. | | |

| Parameter | Permitted Values | Default |
|--|------------------|----------|
| call.singleKeyPressConference¹ | 0 or 1 | 0 |
| <p>If set to 1, the conference will be setup after a user presses the Conference soft key or Conference key the first time. Also, all sound effects (dial tone, DTMF tone while dialing and ringing back) are heard by all existing participants in the conference.</p> <p>If set to 0, sound effects are only heard by conference initiator (original behavior).</p> <p>Note: This feature is supported only for SoundPoint IP 550, 560, and 650. For all others, set to 0.</p> | | |
| call.stickyAutoLineSeize¹ | 0 or 1 | 0 |
| <p>If set to 1, the phone uses <i>sticky</i> line seize behavior. This will help with features that need a second call object to work with. The phone will attempt to initiate a new outgoing call on the same SIP line that is currently in focus on the LCD (this was the behavior in SIP 1.6.5). Dialing through the call list when there is no active call will use the line index for the previous call. Dialing through the call list when there is an active call will use the current active call line index. Dialing through the contact directory will use the current active call line index.</p> <p>If set to 0, the feature is disabled (this was the behavior in SIP 1.6.6). Dialing through the call list will use the line index for the previous call. Dialing through the contact directory will use a random line index.</p> <p>Note: This may fail due to glare issues in which case the phone may select a different available line for the call.</p> | | |
| call.stickyAutoLineSeize.onHookDialing¹ | 0 or 1 | 0 |
| <p>If <code>call.stickyAutoLineSeize</code> is set to 1, this parameter has no effect. The regular <code>stickyAutoLineSeize</code> behavior is followed.</p> <p>If <code>call.stickyAutoLineSeize</code> is set to 0 and this parameter is set to 1, this overrides the <code>stickyAutoLineSeize</code> behavior for hot dial only. (Any new call scenario seizes the next available line.)</p> <p>If <code>call.stickyAutoLineSeize</code> is set to 0 and this parameter is set to 0, there is no difference between hot dial and new call scenarios.</p> <p>Note: A hot dial occurs on the line which is currently in the call appearance. Any new call scenario seizes the next available line.</p> | | |
| call.transferOnConferenceEnd¹ | 0 or 1 | 1 |
| <p>The behavior when the conference host exits a conference. If 0, all parties are disconnected when the conference host exits the conference. If 1, the other parties are left connected when the host exits the conference (the host performs an attended transfer to the other parties, this is the old behavior).</p> | | |
| call.transfer.blindPreferred¹ | 0 or 1 | 0 |
| <p>SoundPoint IP 321, 331, and 335 only. If 1, the blind transfer is the default mode. The Normal soft key is available to switch to a consultative transfer. If 0, the consultative transfer is the default mode. The Blind soft key is available to switch to a blind transfer.</p> | | |
| call.urlModeDialing¹ | 0 or 1 | 0 |
| <p>If 0, URL dialing is disabled. If 1, URL dialing is enabled. Note: URL dialing is supported on SoundPoint IP 321/331/335 phones for unregistered lines only.</p> | | |

¹ Change causes phone to restart or reboot.

<callLists/>

The call lists (or call log) parameter is supported only on VVX 500 and 1500 phones and SpectraLink handsets.

Table 13-9: Call List (Call Log) Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| callLists.collapseDuplicates | 0 or 1 | 1 |
| If 0, all calls are archived and presented in the call lists. If 1, consecutive incomplete between the same party in the same direction (outgoing/incoming) are collapsed into one record with the most recent call displaying. | | |
| callLists.logConsultationCalls | 0 or 1 | 0 |
| If 1, all consultation calls are logged. (Calls made to a third party—while the original party is on hold—when settings up a conference call are called consultation calls.)
If 0, consultation calls are not logged. | | |
| callLists.size | 10 to 99 | 99 |
| The maximum number of retained records of each type (incoming, outgoing, and missed). When the maximum number is reached, new records will overwrite existing records. You can clear the list using the phone's menu system. If you want to prevent the records from uploading to the provisioning server, enter a false URL in the CALL_LISTS_DIRECTORY field in the master configuration file. | | |
| callLists.writeDelay.journal | 1 to 600 | 5 |
| The delay (in seconds) before changes due to an in-progress call are flushed to the file system as a journal. | | |
| callLists.writeDelay.terminated | 10 to 600 | 60 |
| The minimum period between writing out the complete XML file to the local file system and, optionally, to the provisioning server. | | |

<device/>

The <device/> parameters—also known as device settings—contain default values that you can use to configure basic settings for multiple phones.



Web Info: Default Device Parameter Values

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 [Shipping Configuration Notice](#).

Polycom provides a global `device.set` parameter that you can enable for software installation and changes to device parameters. Once you have completed the software installation or made

configuration changes to device parameters, remove `device.set`. Disabling the parameter after the initial software installation prevents the phones from rebooting and triggering a reset of device parameters that users may have changed after the initial installation.

Each `<device/>` parameter has a corresponding `.set` parameter that enables or disables the value for that device parameter. You will need to enable the corresponding `.set` parameter for each parameter you want to apply.



Settings: Each `<device/>` Parameter has a Corresponding `.set` Parameter with One Exception

Note that each `<device/>` parameter has a corresponding `.set` parameter that enables or disables the parameter. There is one exception to this rule: the `device.sec.TLS.customDeviceCertX.set` parameter applies to both `device.sec.TLS.customDeviceCertX.publicCert` and to `device.sec.TLS.customDeviceCertX.privateKey`.

Use Caution When Changing Device Parameters

Use caution when changing `<device/>` parameters as incorrect settings may apply the same IP address to multiple phones.

Note that some parameters may be ignored. For example, if DHCP is enabled it will still override the value set with `device.net.ipAddress`.

Though individual parameters are checked to see whether they are in range, the interaction between parameters is not checked. If a parameter is out of range, an error message will display in the log file and parameter will not be used.

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, Polycom recommends that you test the new configuration files on two phones before initializing all phones.

The following table outlines the three types of `<device/>` parameters, their permitted values, and the default value.

Table 13-10: Device Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| device.set¹ | 0 or 1 | 0 |
| If set to 0, do not use any <code>device.xxx</code> fields to set any parameters. Set this to 0 after the initial software installation. | | |
| If set to 1, use the <code>device.xxx</code> fields that have <code>device.xxx.set=1</code> . Set this to 1 only for the initial software installation. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| device.xxx¹ | string | Null |
| Configuration parameter. | | |
| device.xxx.set¹ | 0 or 1 | 0 |
| If set to 0, do not use the <code>device.xxx</code> value. If set to 1, use the <code>device.xxx</code> value.
For example, if <code>device.net.ipAddress.set=1</code> , then use the value set for <code>device.net.ipAddress</code> . | | |

¹ Change causes phone to restart or reboot.

The following table lists each of the `<device/>` parameters that you can configure.

Table 13-11: Device Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|----------------------------------|----------------|
| device.auth.localAdminPassword | string (32 character max) | Null |
| The phone's local administrative password. The minimum length is defined by sec.pwd.length.admin . | | |
| device.auth.localUserPassword | string (32 character max) | Null |
| The phone user's local password. The minimum length is defined by sec.pwd.length.user . | | |
| device.baseProfile | Generic, Lync | Null |
| Choose the Base Profile that the phone will operate with. | | |
| device.cma.mode | Static, Auto, Disabled | Null |
| The mode the phone uses to retrieve the Polycom CMA server IP address. <i>Auto</i> The phone uses SRV lookup. <i>Disabled</i> The phone does not contact the server. <i>Static</i> The phone uses the server name or IP address specified in <code>device.cma.serverName</code> . <i>Note:</i> If you will modify this parameter, the phone will re-provision. The phone may also reboot if the configuration on the CMA server has changed. | | |
| device.cma.serverName | server name or IP address | Null |
| Polycom CMA server name or IP address. <i>Note:</i> If you will modify this parameter, the phone will re-provision. The phone may also reboot if the configuration on the CMA server has changed. | | |
| device.dhcp.bootSrvOpt¹ | Null, 128 to 254 | Null |
| When the boot server is set to <i>Custom</i> or <i>Custom+Option66</i> , specify the numeric DHCP option that the phone will look for. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|----------------|
| device.dhcp.bootSrvOptType¹ | IP or String | Null |
| The type of DHCP option in which the phone will look for its provisioning server (if <code>device.dhcp.bootSrvUseOpt</code> is set to <code>Custom</code>). If IP, the IP address provided must specify the format of the provisioning server. If String, the string provided must match one of the formats specified by <code>device.prov.serverName</code> . | | |
| device.dhcp.bootSrvUseOpt¹ | Default, Custom, Static, CustomAndDefault | Null |
| <p>Default The phone will look 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 <code>device.prov.serverName</code>.</p> <p>Custom The phone will look for the option number specified by <code>device.dhcp.bootSrvOpt</code>, and the type specified by <code>device.dhcp.bootSrvOptType</code> in the response received from the DHCP server.</p> <p>Static The phone will use the boot server configured through the provisioning server <code>device.prov.*</code> parameters.</p> <p>Custom and Default The phone will use the custom option first or use Option 66 if the custom option is not present.</p> | | |
| device.dhcp.enabled¹ | 0 or 1 | Null |
| If 0, DHCP is disabled. If 1, DHCP is enabled. | | |
| device.dhcp.option60Type¹ | Binary, ASCII | Null |
| 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. | | |
| device.dhcp.dhcpVlanDiscUseOpt¹ | Disabled, Fixed, Custom | Null |
| VLAN Discovery. Disabled , no VLAN discovery through DHCP. Fixed , use predefined DHCP vendor-specific option values of 128, 144, 157 and 191 (<code>device.dhcp.dhcpVlanDiscOpt</code> will be ignored). Custom , use the number specified by <code>device.dhcp.dhcpVlanDiscOpt</code> . | | |
| device.dhcp.dhcpVlanDiscOpt¹ | 128 to 254 | Null |
| The DHCP private option to use when <code>device.dhcp.dhcpVlanDiscUseOpt</code> is set to <code>Custom</code> . | | |
| device.dns.altSrvAddress¹ | server address | Null |
| The secondary server to which the phone directs Domain Name System (DNS) queries. | | |
| device.dns.domain¹ | string | Null |
| The phone's DNS domain. | | |
| device.dns.serverAddress¹ | string | Null |
| The primary server to which the phone directs Domain Name System queries. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| device.em.power¹ | 0 or 1 | Null |
| Applies to expansion modules on IP 650 phones that are powered using Power over Ethernet (PoE). If 0, the phone sets CDP power requirements so expansion modules will not be powered and will not work. If 1, the phone sets CDP power requirements so up to three expansion modules can be powered. | | |
| device.host.hostname¹ | string | Null |
| This parameter enables you to specify a hostname for the phone when using DHCP by adding a hostname string to the phone's configuration. If <code>device.host.hostname.set=1</code> , and <code>device.host.hostname=Null</code> , the DHCP client uses Option 12 to send a predefined hostname to the DHCP registration server using <code>Polycom_<MACaddress></code> . Note that the maximum length of the hostname string is <code><=255</code> bytes. The valid character set is defined in RFC1035. | | |
| device.logincred.domain¹ | string | Null |
| The CMA account domain. <i>Note:</i> If you will modify this parameter, the phone will re-provision. The phone may also reboot if the configuration on the CMA server has changed. | | |
| device.logincred.password¹ | string | Null |
| The CMA account password. <i>Note:</i> If you will modify this parameter, the phone will re-provision. The phone may also reboot if the configuration on the CMA server has changed. | | |
| device.logincred.user¹ | string | Null |
| The CMA account user name. <i>Note:</i> If you will modify this parameter, the phone will re-provision. The phone may also reboot if the configuration on the CMA server has changed. | | |
| device.net.cdpEnabled¹ | 0 or 1 | Null |
| If set to 1, the phone will attempt to determine its VLAN ID and negotiate power through CDP. | | |
| device.net.dot1x.anonid¹ | string | Null |
| EAP-TTLS and EAP-FAST only. The anonymous identity (user name) for 802.1X authentication. | | |
| device.net.dot1x.eapFastInBandProv¹ | 0 or 1 | Null |
| EAP-FAST only, optional. Choose 1 to enable EAP In-Band Provisioning by server unauthenticated PAC provisioning using anonymous Diffie-Hellman key exchange. Choose 0 to disable EAP In-Band Provisioning.
<i>Reserved for Future Use – Choose 2 to enable EAP In-band provisioning by server authenticated PAC provisioning using certificate based server authentication.</i> | | |
| device.net.dot1x.enabled¹ | 0 or 1 | Null |
| If 0, 802.1X authentication is disabled. If 1, 802.1X authentication is enabled. | | |
| device.net.dot1x.identity¹ | string | Null |
| The identity (user name) for 802.1X authentication. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---|----------------|
| device.net.dot1x.method | EAP-None, EAP-TLS, EAP-PEAPv0-MSCHAPv2, EAP-PEAPv0-GTC, EAP-TTLS-MSCHAPv2, EAP-TTLS-GTC, EAP-FAST, EAP-MD5 | Null |
| Specify the 802.1X authentication method, where <code>EAP-NONE</code> means no authentication. | | |
| device.net.dot1x.password¹ | string | Null |
| The password for 802.1X authentication. This parameter is required for all methods except EAP-TLS. | | |
| device.net.ether1000BTClockLAN¹ | Auto, Slave, Master | Null |
| The mode of the LAN clock. Polycom recommends that you do not change this value unless you have Ethernet connectivity issues. | | |
| device.net.ether1000BTClockPC¹ | Auto, Slave, Master | Null |
| The mode of the PC clock. Polycom recommends that you do not change this value unless you have Ethernet connectivity issues. | | |
| device.net.etherModeLAN¹ | Auto, 10HD, 10FD, 100HD, 100FD, 100FD | Null |
| The LAN port mode that sets the network speed over Ethernet. HD means half-duplex and FD means full duplex. <i>Note:</i> Polycom recommends that you do not change this setting. | | |
| device.net.etherModePC¹ | Disabled, Auto, 10HD, 10FD, 100HD, 100FD, 100FD | Auto |
| The PC port mode that sets the network speed over Ethernet. If set to <code>Disabled</code> , the PC port is disabled. HD means half duplex and FD means full duplex. | | |
| device.net.etherStormFilter¹ | 0 or 1 | Null |
| If 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. If 0, DoS Storm Prevention is disabled. | | |
| device.net.etherVlanFilter¹ | 0 or 1 | Null |
| If 1, VLAN filtering is enabled and received Ethernet packets are filtered so the TCP/IP stack doesn't process invalid data or too much data. If 0, VLAN filtering is disabled.
Note that VLAN filtering is not supported on the VVX family of phones. | | |
| device.net.ipAddress¹ | string | Null |
| The phone's IP address. <i>Note:</i> This parameter is disabled when DHCP is enabled (<code>device.dhcp.enabled</code> is set to 1). | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|----------------|
| device.net.IPgateway¹ | dotted-decimal IP address | Null |
| The phone's default router. | | |
| device.net.lldpEnabled¹ | 0 or 1 | Null |
| If set to 1, the phone will attempt to determine its VLAN ID and negotiate power through LLDP. | | |
| device.net.subnetMask¹ | dotted-decimal subnet mask | Null |
| The phone's subnet mask. <i>Note:</i> This parameter is disabled when DHCP is enabled (<code>device.dhcp.enabled</code> is set to 1). | | |
| device.net.vlanId¹ | string | Null |
| The phone's 802.1Q VLAN identifier. If Null, no VLAN tagging. | | |
| device.pacfile.data¹ | String | Null |
| EAP-FAST only, optional. The PAC file (base 64 encoded). To generate a base 64-encoded PAC file, generate the PAC file using your authentication server and then convert it to base 64. You can convert the file to base 64 using the following openssl commands:
<pre>\$ openssl enc -base64 -in myfile -out myfile.b64</pre> | | |
| device.pacfile.password¹ | String | Null |
| EAP-FAST only, optional. The password for the PAC file. | | |
| device.prov.maxRedunServers¹ | 1 to 8 | Null |
| The maximum number of IP addresses that will be used from the DNS. | | |
| device.prov.password¹ | string | Null |
| The password for the phone to log in to the provisioning server. Note that a password may not be required. <i>Note:</i> If you modify this parameter, the phone will re-provision. The phone may also reboot if the configuration on the provisioning server has changed. | | |
| device.prov.redunAttemptLimit¹ | 1 to 10 | Null |
| The maximum number of attempts to attempt a file transfer before the transfer fails. | | |
| device.prov.redunInterAttemptDelay¹ | 0 to 300 | Null |
| The number of seconds to wait after a file transfer fails before retrying the transfer. | | |
| device.prov.serverName | dotted-decimal IP address, domain name string, or URL | Null |
| The IP address, domain name, or URL of the provisioning server, followed by an optional directory and optional configuration filename. This parameter is used if DHCP is disabled (<code>device.dhcp.enabled</code> is 0), if the DHCP server does not send a boot server option, or if the boot server option is static (<code>device.dhcp.bootSrvUseOpt</code> is <code>static</code>). <i>Note:</i> If you modify this parameter, the phone will re-provision. The phone may also reboot if the configuration on the provisioning server has changed. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---|----------------|
| device.prov.serverType¹ | FTP, TFTP, HTTP, HTTPS, FTPS | Null |
| The protocol the phone uses to connect to the provisioning server. <i>Note:</i> Active FTP is not supported for BootROM version 3.0 or later. <i>Note:</i> Only implicit FTPS is supported. | | |
| device.prov.upgradeServer | string | Null |
| The server used by the Polycom Web Configuration Utility's software upgrade feature. The server checks this URL for new software files. | | |
| device.prov.tagSerialNo | 0 or 1 | Null |
| If 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. If 1, the phone's serial number is included. | | |
| device.prov.user | string | Null |
| The user name required for the phone to log in to the provisioning server (if required). <i>Note:</i> If you modify this parameter, the phone will re-provision. The phone may also reboot if the configuration on the provisioning server has changed. | | |
| device.prov.ztpEnabled | 0 or 1 | Null |
| If 0, Disable the ZTP feature. If 1, enable the ZTP feature. | | |
| device.sec.configEncryption.key¹ | string | Null |
| The configuration encryption key used to encrypt configuration files. For more information, see Encrypting Configuration Files. | | |
| device.sec.TLS.customCaCert1 (TLS Platform Profile 1)
device.sec.TLS.customCaCert2 (TLS Platform Profile 2) | string, PEM format | Null |
| 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 <code>device.sec.TLS.profile.caCertList</code> must be configured to use a custom certificate. | | |
| device.sec.TLS.customDeviceCert1.publicCert
device.sec.TLS.customDeviceCert2.publicCert | Enter the signed custom device certificate in PEM format (X.509) | Null |
| device.sec.TLS.customDeviceCert1.publicKey
device.sec.TLS.customDeviceCert2.publicKey | Enter the corresponding signed private key in PEM format (X.509) | Null |
| device.sec.TLS.customDeviceCert1.set
device.sec.TLS.customDeviceCert2.set | 0 or 1 | 0 |
| Note that you use a single <code>.set</code> parameter to enable or disable only these two related <code><device/></code> parameters - <code>device.sec.TLS.customDeviceCertX.publicCert</code> and <code>device.sec.TLS.customDeviceCertX.publicKey</code> . All other <code><device/></code> parameters have their own corresponding <code>.set</code> parameter that will enable or disable that parameter. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|----------------|
| device.sec.TLS.profile.caCertList1 (TLS Platform Profile 1) | Builtin, BuiltinAndPlatform1, BuiltinAndPlatform2, All, Platform1, Platform2, Platform1AndPlatform2 | Null |
| device.sec.TLS.profile.caCertList2 (TLS Platform Profile 2) | | |

Choose the CA certificate(s) to use for TLS Platform Profile 1 and TLS Platform Profile 2 authentication:

- The built-in default certificate
- The built-in and Custom #1 certificates
- The built-in and Custom #2 certificates
- Any certificate (built in, Custom #1 or Custom #2)
- Only the Custom #1 certificate
- Only the Custom #2 certificate
- Either the Custom #1 or Custom #2 certificate

| | | |
|---|---------------|-------------|
| device.sec.TLS.profile.cipherSuite1 (TLS Platform Profile 1) | string | Null |
| device.sec.TLS.profile.cipherSuite2 (TLS Platform Profile 2) | | |

The cipher suites to use for TLS Platform Profile 1 and TLS Platform Profile 2)

| | | |
|--|---------------|-------------|
| device.sec.TLS.profile.cipherSuiteDefault1 (TLS Platform Profile 1) | 0 or 1 | Null |
| device.sec.TLS.profile.cipherSuiteDefault2 (TLS Platform Profile 2) | | |

The cipher suite to use for TLS Platform Profile 1 and TLS Platform profile 2. If set to 0, the custom cipher suite will be used. If set to 1, the default cipher suite will be used.

| | | |
|--|--------------------------------------|-------------|
| device.sec.TLS.profile.deviceCert1 (TLS Platform Profile 1) | Builtin, Platform1, Platform2 | Null |
| device.sec.TLS.profile.deviceCert2 (TLS Platform Profile 2) | | |

Choose the device certificate(s) for TLS Platform Profile 1 and TLS Platform Profile 2 to use for authentication.

| | | |
|--|---|-------------|
| device.sec.TLS.profile.profileSelection.dot1x | PlatformProfile1, PlatformProfile2 | Null |
|--|---|-------------|

Choose the TLS Platform Profile to use for 802.1X, either TLS Platform Profile 1 or TLS Platform Profile 2.

| | | |
|---|---|-------------|
| device.sec.TLS.profileSelection.provisioning¹ | PlatformProfile1, PlatformProfile2 | Null |
|---|---|-------------|

The TLS Platform Profile to use for provisioning, either TLS Platform Profile 1 or TLS Platform Profile 2.

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|----------------|
| device.sec.TLS.profileSelection.syslog¹ | PlatformProfile1,
PlatformProfile2 | Null |
| The TLS Platform Profile to use for syslog, either TLS Platform Profile 1 or TLS Platform Profile 2. | | |
| device.sec.TLS.prov.strictCertCommonNameValidation | 0 or 1 | 1 |
| If set to 1, provisioning always verifies the server certificate for commonName/SubjectAltName match with the server hostname that the phone is trying to connect. | | |
| device.sec.TLS.syslog.strictCertCommonNameValidation | 0 or 1 | 1 |
| If set to 1, syslog always verifies the server certificate for commonName/SubjectAltName match with the server hostname that the phone is trying to connect. | | |
| device.snmp.gmtOffset | -43200 to 46800 | Null |
| The GMT offset – in seconds – to use for daylight savings time, corresponding to -12 to +13 hours. | | |
| device.snmp.serverName | dotted-decimal IP
address or domain
name string | Null |
| The SNMP server from which the phone will obtain the current time. | | |
| device.syslog.facility | 0 to 23 | Null |
| A description of what generated the log message. For more information, see section 4.1.1 or RFC 3164 . | | |
| device.syslog.prependMac¹ | 0 or 1 | Null |
| If 1, the phone's MAC address is pre-pended to the log message sent to the syslog server. | | |
| device.syslog.renderLevel¹ | 0 to 6 | Null |
| Specify the logging level that will display in the syslog. Note that when you choose a log level, you are including all events of an equal or greater severity level and excluding events of a lower severity level. The logging level you choose determines the lowest severity of events that will be logged.
0 or 1: SeverityDebug(7). 2 or 3: SeverityInformational(6). 4: SeverityError(3). 5: SeverityCritical(2). 6: SeverityEmergency(0). | | |
| device.syslog.serverName | dotted-decimal IP
address OR domain
name string | Null |
| The syslog server IP address or domain name string. | | |
| device.syslog.transport | None, UDP, TCP, TLS | Null |
| The transport protocol that the phone will use to write to the syslog server. If set to None, transmission is turned off but the server address is preserved. | | |
| device.usbnet.dhcpServerEnabled | 0 or 1 | Null |
| SpectralLink 8400 Series handsets only. If 1, a DHCP Server (which gives out addresses) needs to be started, as opposed to a DHCP Client (which gets an address). | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|--------------------|
| device.usbnet.ipGateway¹ | String | 169.254.1.1 |
| SpectralLink 8400 Series handsets only. The provisioning server IP address. | | |
| device.usbnet.subnetMask¹ | String | 255.255.0.0 |
| SpectralLink 8400 Series handsets only. The handset's subnet mask for USBNet. | | |
| device.usbnet.enabled¹ | 0 or 1 | 1 |
| SpectralLink 8400 Series handsets only. If 0, USBNet is disabled. If 1, USBNet is enabled. | | |
| device.usbnet.ipAddress¹ | String | 169.254.1.2 |
| SpectralLink 8400 Series handsets only. The handset's dotted-decimal IP address on the USBNet interface. | | |
| device.wifi.ccxMandatory | 0 or 1 | 0 |
| SpectralLink 8400 Series handsets only. If 0, the SpectraLink handsets will connect to access points (APs) that do not advertise Cisco Compatible Extensions (CCX v4) or higher. If 1, the handsets will not connect to APs that do not advertise CCX v4 or higher (CCX is mandatory). | | |
| device.wifi.dhcpEnabled | 0 or 1 | 1 |
| SpectralLink 8400 Series handsets only. If 0, DHCP is disabled on the wireless interface. If 1, DHCP is enabled on the wireless interface. | | |
| device.wifi.dot11n.enabled | 0 or 1 | 0 |
| SpectralLink 8400 Series handsets only. If 0, 802.11n support is disabled. If 1, 802.11n support is enabled. | | |
| device.wifi.enabled | 0 or 1 | 0 |
| SpectralLink 8400 Series handsets only. If 0, the wireless interface is disabled. If 1, the wireless interface is enabled. | | |
| device.wifi.ipAddress | String | 0.0.0.0 |
| SpectralLink 8400 Series handsets only. The IP address of the wireless interface (if not using DHCP). | | |
| device.wifi.ipGateway | String | 0.0.0.0 |
| SpectralLink 8400 Series handsets only. The IP gateway address for the wireless interface (if not using DHCP). | | |
| device.wifi.psk.keyType | 0 or 1 | 0 |
| The key type: key or passphrase. | | |
| device.wifi.psk.key | string | 0xFF |
| The hexadecimal key or ASCII passphrase. | | |
| SpectralLink 8400 Series handsets only. The WPA(2) PSK key type and key. If the key type is 0, a 256-bit hexadecimal key is used. If the key type is 1, a string of 8 to 63 ASCII characters is used as the pass code. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-----------------------------|----------------|
| device.wifi.qos.acMandatory | String | Null |
| SpectraLink 8400 Series handsets only. If 1, the handset will only connect to access points that enforce admission control or access control. If 0, the handset access control or admission control is not necessary. | | |
| device.wifi.radio.band5GHz.subBand1.enable¹ | 0 or 1 | 0 |
| device.wifi.radio.band5GHz.subBand2.enable¹ | 0 or 1 | 0 |
| device.wifi.radio.band5GHz.subBand3.enable¹ | 0 or 1 | 0 |
| device.wifi.radio.band5GHz.subBand4.enable¹ | 0 or 1 | 0 |
| SpectraLink 8400 Series handsets only. If 0, the 5GHz sub-band (sub band 1, 2, 3, or 4) is disabled. If 1, the sub band is enabled. <i>Note:</i> Regulatory authorities (FCC North America) further subdivide the 5GHz band into multiple sub-bands (some of which are not available in all countries). You can enable and disable individual sub-bands and set the maximum transmit power for each. For maximum performance, you should enable the same bands and sub-bands as configured on your wireless infrastructure, otherwise your handset will waste time looking for a signal on the unused sub-bands. | | |
| device.wifi.radio.band5GHz.subBand1.txPower¹ | 1 to 7 | 5 |
| device.wifi.radio.band5GHz.subBand2.txPower¹ | 1 to 7 | 5 |
| device.wifi.radio.band5GHz.subBand3.txPower¹ | 1 to 7 | 5 |
| device.wifi.radio.band5GHz.subBand4.txPower¹ | 1 to 7 | 5 |
| SpectraLink 8400 Series handsets only. The maximum power that the handset will use to transmit in the sub-band (for sub-band 1, 2, 3, and 4). In general, this power should match the power setting at the access point so that the coverage radius of the phone matches that of the access point. | | |
| device.wifi.radio.band5GHzEnable¹ | 0 or 1 | 0 |
| SpectraLink 8400 Series handsets only. If 0, the 5 GHz wireless band is disabled. If 1, the 5 GHz band is enabled. <i>Note:</i> enable the individual sub-bands and set the transmit power for the sub-bands by configuring <code>device.wifi.radio.band5GHz.subBandx</code> . | | |
| device.wifi.radio.band2_4GHzEnable¹ | 0 or 1 | 0 |
| SpectraLink 8400 Series handsets only. If 0, the 2.4 GHz wireless band is disabled. If 1, the 2.4 GHz band is enabled. | | |
| device.wifi.radio.band2_4GHz.txPower¹ | 1 to 7 | 5 |
| SpectraLink 8400 Series handsets only. The maximum power that the handset will use to transmit in the 2.4 GHz band. In general, this power should match the power setting at the access point so that the coverage radius of the phone matches that of the access point. | | |
| device.wifi.radio.regulatoryDomain | 1, 2, 4, 7, 8, or 10 | Null |
| SpectraLink 8400 Series handsets only. Available values specify the regulatory domain. The supported values are 1 (North America), 2 (Europe), 4 (Singapore), 7 (Hong Kong), 8 (Mexico), and 10 (Australia). If Null, no regulatory domain is selected. You must set the regulatory domain before the handsets can be used. There is no default setting for this option and the handsets will not associate with an access point (AP) until you specify a value. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|------------------|
| device.wifi.securityMode¹ | None, WEP, WPA-PSK, WPA2-PSK, WPA2-Enterprise | Null |
| SpectraLink 8400 Series handsets only. The wireless security mode. | | |
| device.wifi.ssid¹ | String | SSID1 |
| SpectraLink 8400 Series handsets only. The Service Set Identifier (SSID) of the wireless network. | | |
| device.wifi.subnetMask¹ | String | 255.0.0.0 |
| SpectraLink 8400 Series handsets only. The network mask address of the wireless interface (if not using DHCP). | | |
| device.wifi.wep.authType¹ | OpenSystem, SharedKey | 0 |
| SpectraLink 8400 Series handsets only. The Wi-Fi WEP authentication type. | | |
| device.wifi.wep.defaultKey¹ | 1 to 4 | 1 |
| SpectraLink 8400 Series handsets only. Specifies which of the four keys from <code>device.wifi.wep.key1</code> to <code>device.wifi.wep.key4</code> is used. | | |
| device.wifi.wep.encryptionEnable¹ | 0 or 1 | 1 |
| SpectraLink 8400 Series handsets only. If 0, WEP encryption is disabled. If 1, WEP encryption is enabled. | | |
| device.wifi.wep.keyLength¹ | 0 or 1 | 0 |
| SpectraLink 8400 Series handsets only. The length of the hexadecimal WEP key. 0 = 40-bits, 1 = 104-bits. | | |
| device.wifi.wep.key1¹ | String | 0xFF |
| device.wifi.wep.key2¹ | String | 0xFF |
| device.wifi.wep.key3¹ | String | 0xFF |
| device.wifi.wep.key4¹ | String | 0xFF |
| SpectraLink 8400 Series handsets only. The WEP hexadecimal key with a 40-bit or 104-bit length, as specified by <code>device.wifi.wep.keyLength</code> . | | |
| device.wifi.wpa2Ent.eapFast.inBandProv¹ | 0 or 1 | 0 |
| SpectraLink 8400 Series handsets only. If 0, the PAC file is initially loaded into to the handset during configuration (called <i>out-of-band</i>). If 1, the PAC file is automatically loaded form the network (called <i>in-band</i>). | | |
| device.wifi.wpa2Ent.method¹ | EAP-PEAPv0/MSCHAPv2, EAP-FAST | Null |
| SpectraLink 8400 Series handsets only. The Extensible Authentication Protocol (EAP) to use for 802.1X authentication. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|-----------------|
| device.wifi.wpa2Ent.password¹ | String | PlcmSplp |
| SpectralLink 8400 Series handsets only. The WPA2-Enterprise password. | | |
| device.wifi.wpa2Ent.roaming¹ | OKC, CCKM | 0 |
| SpectralLink 8400 Series handsets only. The WPA2-Enterprise fast roaming method. If OKC , Opportunistic Key Caching (OKC) is used. If CCKM , Cisco Centralized Key Management (CCKM) is used. The fast roaming methods allow part of the key derived from the server to be cached in the wireless network to shorten the time it takes to renegotiate a secure handoff. | | |
| device.wifi.wpa2Ent.user¹ | String | PlcmSplp |
| SpectralLink 8400 Series handsets only. The WPA2-Enterprise user name. | | |

¹ Change causes phone to restart or reboot.

<dialplan/>

The dial plan (or digit map) is not applied against Placed Call List, Voicemail, last call return, remote control dialed numbers, or on-hook dialing.

This parameter allows the user to create a specific routing path for outgoing SIP calls independent of other *default* configurations.

Table 13-12: Dial Plan (Digit Map) Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| dialplan.applyToCallListDial¹ | 0 or 1 | 1 |
| If 0, the dial plan does not apply to numbers dialed from the Received Call List or Missed Call List. If 1, the dial plan is applied to numbers dialed from the received call and missed call lists, including sub-menus. | | |
| dialplan.applyToDirectoryDial¹ | 0 or 1 | 0 |
| If 0, the dial plan is not applied to numbers dialed from the directory or speed dial list. If 1, the dial plan is applied to numbers dialed from the directory or speed dial, including auto-call contact numbers. | | |
| dialplan.applyToForward¹ | | |
| If 0, the dial plan does not apply to forwarded calls. If 1, the dial plan applies to forwarded calls. | | |
| dialplan.applyToTelUriDial¹ | 0 or 1 | 1 |
| If 0, the dial plan does not apply to URI dialing. If 1, the dial plan applies to URI dialing. | | |
| dialplan.applyToUserDial¹ | 0 or 1 | 1 |
| If 0, the dial plan does not apply to calls made when the user presses the Dial soft key to place a call. If 1, the dial plan applies to calls placed using the Dial soft key. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|--|
| dialplan.applyToUserSend¹ | 0 or 1 | 1 |
| <p>If 0, the dial plan does not apply to calls placed when the user presses the Send soft key to place a call. If 1, the dial plan applies to calls placed using the Send soft key.</p> | | |
| dialplan.digitmap¹ | string compatible with the digit map feature of MGCP described in 2.1.5 of RFC 3435 | [2-9]11 0T +011xxx.T 0[2-9]xxxxxxxx +1[2-9]xxxxxxxx [2-9]xxxxxxxx [2-9]xxxT |
| <p>The digit map used for the dial plan. The string is limited to 2560 bytes and 100 segments of 64 bytes; a comma is also allowed; a comma will turn dial tone back on; '+' is allowed as a valid digit; extension letter 'R' is used as defined above. This parameter enables the phone to automatically initiate calls to numbers that match a digit map pattern.</p> | | |
| dialplan.digitmap.timeOut¹ | string of positive integers separated by ' ' | 3 3 3 3 3 |
| <p>Specify a timeout in seconds for each segment of digit map. After you press a key, the phone will wait this many seconds before matching the digits to a dial plan and dialing the call. <i>Note:</i> If there are more digit maps than timeout values, the default value of 3 will be used. If there are more timeout values than digit maps, the extra timeout values are ignored.</p> | | |
| dialplan.filterNonDigitUriUsers¹ | 0 or 1 | 0 |
| <p>If 0, allow do not filter out (+) in the dial plan. If 1, filter out (+) from the dial plan (this is the previous behavior).</p> | | |
| dialplan.impossibleMatchHandling¹ | 0, 1 or 2 | 0 |
| <p>This parameter applies to digits entered in dial mode. Users are in dial mode after having picked up the handset, headset, or pressed the New Call key, and not when hot dialing, contact dialing, or call list dialing. If set to 0, the digits entered up to and including the point where an impossible match occurred are sent to the server immediately. If set to 1, give reorder tone. If set to 2, allow user to accumulate digits and dispatch call manually with the Send soft key.</p> | | |
| dialplan.removeEndOfDial¹ | 0 or 1 | 1 |
| <p>If set to 1, strip trailing # digit from digits sent out.</p> | | |
| dialplan.routing.emergency.outboundIdentity | SIP, secure SIP, or TEL URI | Null |
| <p>The identity used to identify your phone when you place an emergency call from your phone. A valid SIP, secure SIP, or TEL URI. The string may be 10 to 25 characters in length.</p> | | |

| Parameter | Permitted Values | Default |
|--|------------------------|--------------------------------|
| dialplan.routing.emergency.x.description¹
Emergency contact description | string | x=1:Emergency,
Others: Null |
| dialplan.routing.emergency.x.server.y¹
Emergency server | positive integer | x=1: 1, others: Null |
| dialplan.routing.emergency.x.value
Emergency URL values | SIP URL (single entry) | x=1: 911, others: Null |

x is the index of the emergency entry description and y is the index of the server associated with emergency entry x. For each emergency entry (index x), one or more server entries (indexes (x,y)) can be configured. x and y must both use sequential numbering starting at 1.

`description`: The label or description for the emergency address

`server.y`: The index representing the server to use for emergency routing (dialplan.routing.server.x.address where x is the index).

`value`: The URLs that should be watched for. When the user dials one of the URLs, the call will be directed to the emergency server defined by `address`.

Note: Blind transfer for 911 (or other emergency calls) may not work if registration and emergency servers are different entities.

| | | |
|--|--|------|
| dialplan.routing.server.x.address¹ | dotted-decimal IP
address or hostname | Null |
|--|--|------|

The IP address or hostname of a SIP server that will be used for routing calls. Multiple servers can be listed starting with x=1 to 3 for fault tolerance. **Note:** Blind transfer for 911 (or other emergency calls) may not work if registration and emergency servers are different entities.

| | | |
|---|------------|------|
| dialplan.routing.server.x.port¹ | 1 to 65535 | 5060 |
|---|------------|------|

The port of a SIP server that will be used for routing calls

| | | |
|--|--|----------|
| dialplan.routing.server.x.transport¹ | DNSnaptr,
TCPpreferred,
UDPOnly, TLS,
TCPOnly | DNSnaptr |
|--|--|----------|

The dns lookup of the first server to be dialed will be used, if there is a conflict with the others.

For example, if `dialplan.routing.server.1.transport="UDPOnly"` and `dialplan.routing.server.2.transport = "TLS"`, then UDPOnly is used.

¹ Change causes phone to restart or reboot.

Per-registration dial plan configuration is also supported. The descriptions for each parameter are in the table above. The parameters listed in this table override the parameters in the previous table for registration x, where x is the registration number (for example, `dialplan.x.applyToTelUriDial` overrides `dialplan.applyToTelUriDial` for registration x):
For IP 321/331/335: x=1-2; IP 450: x=1-3; IP 550, 560: x=1-4; VVX 500:x=1-12; VVX 1500: x=1-6; IP 650; IP 5000, IP 6000, IP 7000: x=1.

Table 13-13: Per-Registration Dial Plan (Digit Map) Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|---------------------------------------|
| dialplan.x.applyToCallListDial ¹ | 0 or 1 | 1 |
| dialplan.x.applyToDirectoryDial ¹ | 0 or 1 | 0 |
| dialplan.x.applyToForward | 0 or 1 | 0 |
| dialplan.x.applyToTelUriDial ¹ | 0 or 1 | 1 |
| dialplan.x.applyToUserDial ¹ | 0 or 1 | 1 |
| dialplan.x.applyToUserSend ¹ | 0 or 1 | 1 |
| dialplan.x.digitmap ¹ | string - max number of characters 2560 | Null |
| dialplan.x.digitmap.timeOut ¹ | string - max number of characters 100 | Null |
| dialplan.x.e911dialmask | string - max number of characters 256 | Null |
| dialplan.x.e911dialstring | string - max number of characters 256 | Null |
| dialplan.x.applyToForward | 0 or 1 | 0 |
| dialplan.x.impossibleMatchHandling ¹ | 0 to 2 | 0 |
| dialplan.x.originaldigitmap | string - max number of characters 2560 | Null |
| dialplan.x.removeEndOfDial ¹ | 0 or 1 | 1 |
| dialplan.x.routing.emergency.y.value ¹ | string - max number of characters 64 | Null |
| dialplan.x.routing.emergency.y.server.z ¹ | 0 to 3 | 0 For all x, y, and z = 1 to 3 |
| dialplan.x.routing.server.y.address ¹ | string - max number of characters 256 | Null |
| dialplan.x.routing.server.y.port ¹ | 1 to 65535 | 5060 |
| dialplan.x.routing.server.y.transport ¹ | DNSnaptr, TCPpreferred, UDPOnly, TLS, TCPOnly | DNSnaptr |

¹ Change causes phone to restart or reboot.

<dir>

This parameter definition includes:

- <local/> - The local directory definition
- <corp/>- The corporate directory definition

<local/>

The local directory is stored in either device settings or RAM on the phone. The local directory size is limited based on the amount of flash memory in the phone. (Different phone models have variable flash memory.)

When the volatile storage option is enabled, ensure that a properly configured provisioning server that allows uploads is available to store a back-up copy of the directory or its contents will be lost when the phone reboots or loses power.

Table 13-14: Local Contact Directory Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|--------------------------|
| dir.local.contacts.maxNum¹ | IP 321, 331, 335, and 7000: 1 to 99
Other Phones: 1 to 9999 | 99
9999 |
| Maximum number of contacts allowed in the local contact directory. | | |
| dir.local.readonly¹ | 0 or 1 | 0 |
| If 0, the local contact directory can be edited. If 1, the local contact directory is read-only.
<i>Note:</i> If 1 (read only), speed dial entry on the SoundPoint IP 321/331/335 is disabled (enter the speed dial index followed by #). | | |
| dir.search.field | 0 or 1 | 0 |
| If 0, search the contact directory by contact's last name. If 1, search by first name. | | |

¹ Change causes phone to restart or reboot.

<corp/>

A portion of the corporate directory is stored in flash memory on the phone. The size is based on the amount of flash memory in the phone. (Different phone models have variable flash memory.)

Table 13-15: Corporate Directory Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---|------------------|
| dir.corp.address¹ | dotted-decimal IP address or hostname or FQDN | Null |
| The IP address or hostname of the LDAP server interface to the corporate directory. For example, <i>host.domain.com</i> . | | |
| dir.corp.attribute.x.filter¹ | UTF-8 encoded string | Null |
| The filter string for this parameter, which is edited when searching. | | |
| dir.corp.attribute.x.label¹ | UTF-8 encoded string | Null |
| The label when data is displayed. | | |
| dir.corp.attribute.x.name¹ | UTF-8 encoded string | Null |
| The name of the parameter to match on the server. Each name must be unique; however, an LDAP entry can have multiple parameters with the same name. Up to eight parameters can be configured (x = 1 to 8). | | |
| dir.corp.attribute.x.searchable¹ | 0 or 1 | 0 |
| If 0, quick search on parameter x (if x is 2 or more) is disabled. If 1, quick search on x (if x is 2 or more) is enabled. | | |
| dir.corp.attribute.x.sticky¹ | 0 or 1 | 0 |
| If 0, the filter criteria for attribute x is reset after a reboot. If 1, the filter criteria are retained through a reboot. If you set an attribute to be sticky (set this parameter to 1), a '*' will display before the label of the attribute on the phone. | | |
| dir.corp.attribute.x.type¹ | first_name, last_name, phone_number SIP_address, H323_address URL, other | last_name |
| Defines how parameter x is interpreted by the phone. Entries can have multiple parameters of the same type. The value other is used for display purposes only.
If the user saves the entry to the local contact directory on the phone, <i>first_name</i> , <i>last_name</i> , and <i>phone_number</i> are copied. The user can place a call to the <i>phone_number</i> and <i>SIP_address</i> from the corporate directory. | | |
| dir.corp.autoQuerySubmitTimeout¹ | 0 to 60 seconds | 0 |
| The timeout (in seconds) between when the user stops entering characters in the quick search and when the search query is automatically submitted. If 0, there is no timeout (automatic submit is disabled). | | |
| dir.corp.backGroundSync¹ | 0 or 1 | 0 |
| If 0, background downloading from the LDAP server is disabled. If 1, background downloading is enabled. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-----------------------------|-----------------------------|
| dir.corp.backGroundSync.period¹ | 3600 to 604800 | 86400 |
| The corporate directory cache is refreshed after the corporate directory feature has not been used for this period of time seconds. The default period is 24 hours (86400 seconds). The minimum is 1 hour and the maximum is 7 days. | | |
| dir.corp.baseDN¹ | UTF-8 encoded string | Null |
| The base domain name. This is the starting point for making queries on the LDAP server. | | |
| dir.corp.bindOnInit¹ | 0 or 1 | 1 |
| If 0, do not use bind authentication on initialization. If 1, use bind authentication on initialization. | | |
| dir.corp.cacheSize¹ | 8 to 256 | 128 |
| The maximum number of entries that can be cached locally on the phone. | | |
| dir.corp.filterPrefix¹ | UTF-8 encoded string | (objectclass=person) |
| Predefined filter string for search queries. | | |
| dir.corp.pageSize¹ | 8 to 64 | 32 |
| The maximum number of entries requested from the corporate directory server with each query. | | |
| dir.corp.password¹ | UTF-8 encoded string | Null |
| The password used to authenticate to the LDAP server. | | |
| dir.corp.port¹ | 0, Null, 1 to 65535 | 389 (TCP) 636 (TLS) |
| The port that connects to the server if a full URL is not provided. | | |
| dir.corp.scope¹ | one, sub, base | sub |
| The type of search that is performed. If one , a search of one level below the base domain name (DN). If sub , a recursive search of all levels below the base DN. If base , a search at the base DN level. | | |
| dir.corp.sortControl¹ | 0 or 1 | 0 |
| Control how a client can make queries and sorts entries locally. If 0, leave sorting as negotiated between the client and server. If 1, force sorting of queries (this causes excessive LDAP queries and should only be used to diagnose LDAP servers with sorting problems). | | |
| dir.corp.transport¹ | TCP, TLS, Null | TCP |
| Specify whether a TCP or TLS connection is made with the server, if a full URL is not provided. | | |
| dir.corp.user¹ | UTF-8 encoded string | Null |
| The user name used to authenticate to the LDAP server. | | |
| dir.corp.viewPersistence¹ | 0 or 1 | 0 |
| If 0, the corporate directory search filters and browsing position are reset each time the user accesses the corporate directory. If 1, the search filters and browsing position from the previous session are displayed each time the user accesses the corporate directory. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---------------------------|----------------|
| dir.corp.vlv.allow¹ | 0 or 1 | 0 |
| If 0, virtual view list (VLV) queries are disabled. If 1, VLV queries are enabled and can be made if the LDAP server supports VLV. | | |
| dir.corp.vlv.sortOrder¹ | list of parameters | Null |
| The list of parameters—in exact order—for the LDAP server to use when indexing. For example: <code>sn</code> , <code>givenName</code> , <code>telephoneNumber</code> . | | |

¹ Change causes phone to restart or reboot.

<divert/>

The phone has a flexible call forward/diversion feature for each registration. In all cases, a call will only be diverted if a non-Null contact has been configured.

In the following table, x is the registration number. IP 321/331/335: x=1-2; IP 450: x=1-3; IP 550, 560: x=1-4; VVX 500:x=1-12; VVX 1500: x=1-6; IP 650: x=1-34; IP 5000: x=1; IP 6000: x=1; SL8400: x=1-6.

Table 13-16: Call Diversion (Call Forwarding) Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---|----------------|
| divert.x.contact¹ | contact address: 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 |
| The forward-to contact used for all automatic call diversion features. All automatically forwarded calls will be directed to this contact. The contact can be overridden by a busy contact, DND contact, or no-answer contact as specified by the <code>busy</code> , <code>dnd</code> , and <code>noAnswer</code> parameters that follow. | | |
| divert.x.sharedDisabled¹ | 0 or 1 | 1 |
| If 0, call diversion features can be used on shared lines. If 1, call diversion features are disabled on shared lines. | | |
| divert.x.autoOnSpecificCaller² | 0 or 1 | 1 |
| If 0, the Auto Divert feature of the contact directory is disabled for registration x. If 1, calls on registration x may be diverted using Auto Divert, you may specify to divert individual calls or divert all calls. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| divert.busy.x.enabled² | 0 or 1 | 1 |
| divert.busy.x.contact¹ | contact address | Null |
| Divert incoming calls that reach a busy signal. If <i>enabled</i> is set to 1, calls will be diverted when registration <i>x</i> is busy. Calls will be sent to the busy contact's address if it is specified; otherwise calls will be sent to the default contact specified by <i>divert.x.contact</i> . If <i>enabled</i> is set to 0, calls will not be diverted if the line is busy. | | |
| divert.dnd.x.enabled² | 0 or 1 | 0 |
| divert.dnd.x.contact¹ | contact address | Null |
| Divert calls when Do Not Disturb is enabled. If <i>enabled</i> is set to 1, calls will be diverted when DND is enabled on registration <i>x</i> . Calls will be sent to the DND contact's address if it is specified; otherwise calls will be sent to the default contact specified by <i>divert.x.contact</i> . | | |
| divert.fwd.x.enabled² | 0 or 1 | 1 |
| If 0, the user cannot enable universal call forwarding (automatic forwarding for all calls on registration <i>x</i>). If 1, a <i>Forward</i> soft key displays on the phone's Home screen that you can use to enable universal call forwarding. If enabled on SpectraLink 84xx Series wireless handset, a <i>Forward</i> soft key will display on the flyout menu when you press the <i>Features</i> soft key. | | |
| divert.noanswer.x.enabled² | 0 or 1 | 1 |
| divert.noanswer.x.contact¹ | contact address | Null |
| divert.noanswer.x.timeout¹ | positive integer | 55 |
| If no-answer call diversion is <i>enabled</i> , calls that are not answered after the number of seconds specified by <i>timeout</i> will be sent to the no-answer <i>contact</i> . If the no-answer <i>contact</i> is set to Null, the call will be sent to the default contact specified by <i>divert.x.contact</i> . If <i>enabled</i> is set to 0, calls will not be diverted if they are not answered. | | |

¹ Change causes phone to restart or reboot.

² Change causes phone to restart or reboot. If server-based call forwarding is enabled, this parameter is disabled.

<dns/>

In the tables below, a maximum of 12 [DNS-A](#), [DNS-NAPTR](#), and [DNS-SRV](#) record entries can be added.

DNS-A

Add up to 12 DNS-A record entries using the parameters in [Table 14-17](#). Specify the address, name, and cache time interval for DNS-A record *x*, where *x* is from 1 to 12.

Table 13-17: DNA-A Parameters

| <i>Parameter</i> | <i>Permitted values</i> | <i>Default</i> |
|--|---|----------------|
| dns.cache.A.x.address | dotted-decimal IP version 4 address | Null |
| IP address. | | |
| dns.cache.A.x.name | valid hostname | Null |
| Hostname | | |
| dns.cache.A.x.ttl | 300 to 536870912 (2²⁹), seconds | 300 |
| The TTL describes the time period the phone will use 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 will retry a dynamic network request before falling back on the static entry and its reset TTL timer again. | | |

DNS-NAPTR

Add up to 12 DNS-NAPTR record entries using the parameters in [Table 14-18](#). Specify each parameter for DNS-NAPTR record *x*, where *x* is from 1 to 12.

Table 13-18: DNS-NAPTR Parameters

| <i>Parameter</i> | <i>Permitted values</i> | <i>Default</i> |
|---|---|----------------|
| dns.cache.NAPTR.x.flags | A single character from [A-Z, 0-9] | Null |
| 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. | | |
| dns.cache.NAPTR.x.name | domain name string | Null |
| The domain name to which this resource record refers. | | |
| dns.cache.NAPTR.x.order | 0 to 65535 | 0 |
| An integer specifying the order in which the NAPTR records must be processed to ensure the correct ordering of rules. | | |
| dns.cache.NAPTR.x.preference | 0 to 65535 | 0 |
| 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. | | |

| <i>Parameter</i> | <i>Permitted values</i> | <i>Default</i> |
|---|--|----------------|
| dns.cache.NAPTR.x.regexp | string containing a substitution expression | Null |
| <p>This parameter is currently unused.</p> <p>Applied to the original string held by the client. The substitution expression is applied in order to construct the next domain name that will be looked up. The grammar of the substitution expression is given in RFC 2915.</p> | | |
| dns.cache.NAPTR.x.replacement | domain name string with SRV prefix | Null |
| <p>The next name to query for NAPTR records depending on the value of the flags field. It must be a fully qualified domain-name.</p> | | |
| dns.cache.NAPTR.x.service | string | Null |
| <p>Specifies the service(s) available down this rewrite path. For more information, see RFC 2915.</p> | | |
| dns.cache.NAPTR.x.ttl | 300 to 536870912 (2²⁹), seconds | 300 |
| <p>The TTL describes the time period the phone will use 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 will retry a dynamic network request before falling back on the static entry and its reset TTL timer again.</p> | | |

DNS-SRV

Add up to 12 DNS-SRV record entries using the parameters in [Table 14-19](#). Specify each parameter for DNS-SRV record *x*, where *x* is from 1 to 12.

Table 13-19: DNS-SRV Parameters

| <i>Parameter</i> | <i>Permitted values</i> | <i>Default</i> |
|--|---|----------------|
| dns.cache.SRV.x.name | domain name string with SRV prefix | Null |
| <p>The domain name string with SRV prefix.</p> | | |
| dns.cache.SRV.x.port | 0 to 65535 | 0 |
| <p>The port on this target host of this service. For more information, see RFC 2782.</p> | | |
| dns.cache.SRV.x.priority | 0 to 65535 | 0 |
| <p>The priority of this target host. For more information, see RFC 2782.</p> | | |
| dns.cache.SRV.x.target | domain name string | Null |
| <p>The domain name of the target host. For more information, see RFC 2782.</p> | | |

| <i>Parameter</i> | <i>Permitted values</i> | <i>Default</i> |
|---|---|----------------|
| dns.cache.SRV.x.ttl | 300 to 536870912 (2^29), seconds | 300 |
| <p>The TTL describes the time period the phone will use 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 will retry a dynamic network request before falling back on the static entry and its reset TTL timer again.</p> | | |
| dns.cache.SRV.x.weight | 0 to 65535 | 0 |
| <p>A server selection mechanism. For more information, see RFC 2782.</p> | | |

<efk/>

Use the following three tables to configure the Enhanced Feature Key feature on your phone.

Table 13-20: Enhanced Feature Key (EFK) Parameters

| <i>Parameter Name</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--------------------------------------|----------------|
| efk.version | 2 (1 for SIP 3.0 and earlier) | 2 |
| <p>The version of the EFK elements. For SIP 3.0.x or earlier, 1 is the only supported version. For SIP 3.1 and later, 2 is the only supported version. If this parameter is Null, the EFK feature is disabled. This parameter is not required if there are no <code>efk.efklist</code> entries.</p> | | |

The EFK List parameters are outlined in [Table 14-21: Enhanced Feature Key \(EFK\) List Parameters](#).

Table 13-21: Enhanced Feature Key (EFK) List Parameters

| <i>Parameter Name</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| efk.efklist.x.action.string | | |
| <p>The action string contains a macro definition of the action that the feature key will perform. If EFK is enabled, this parameter must have a value (it cannot be Null). For a list of macro definitions and example macro strings, see Understanding Macro Definitions.</p> | | |
| efk.efklist.x.label | string | Null |
| <p>The text string that will be used as a label on any user text entry screens during EFK operation. If Null, the <i>Null</i> string is used. <i>Note:</i> If the label does not fit on the screen, the text will be shortened and '...' will be appended.</p> | | |

| <i>Parameter Name</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|-----------------------|
| efk.efklist.x.mname | | expanded_macro |
| 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. | | |
| efk.efklist.x.status | 0 or 1 | 0 |
| If 0 or Null, key x is disabled. If 1, the key is enabled. | | |
| efk.efklist.x.type | | invite |
| The SIP method to be performed. If set to <i>invite</i> , the action required is performed using the SIP INVITE method. <i>Note:</i> This parameter is included for backwards compatibility. Do not use if possible. If <i>efk.x.action.string</i> contains types, this parameter is ignored. If Null, the default of INVITE is used. | | |

The EFK Prompt parameters are listed in [Table 14-22: Enhanced Feature Key \(EFK\) Prompt Parameters](#).

Table 13-22: Enhanced Feature Key (EFK) Prompt Parameters

| <i>Parameter Name</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--------------------------|----------------|
| efk.efkprompt.x.label¹ | string | Null |
| The prompt text that is presented to the user on the user prompt screen. If Null, no prompt displays. <i>Note:</i> If the label does not fit on the screen, the label will be shortened and ‘...’ will be appended. | | |
| efk.efkprompt.x.status¹ | 0 or 1 | 0 |
| If 0, key x is disabled. If 1, the key is enabled. This parameter must have a value, it cannot be Null. <i>Note:</i> If a macro attempts to use a prompt that is disabled or invalid, the macro execution will fail. | | |
| efk.efkprompt.x.type¹ | numeric or text | text |
| The type of characters entered by the user. If set to <i>numeric</i> , the characters are interpreted as numbers. If set to <i>text</i> , the characters are interpreted as letters. If Null, <i>numeric</i> is used. If this parameter has an invalid value, this prompt, and all parameters depending on this prompt, are invalid. <i>Note:</i> A mix of <i>numeric</i> and <i>text</i> is not supported. | | |
| efk.efkprompt.x.userfeedback¹ | visible or masked | visible |
| The user input feedback method. If set to <i>visible</i> , the text is visible. If set to <i>masked</i> , the text displays as asterisk characters (*), this can be used to mask password fields. If Null, <i>visible</i> is used. If this parameter has an invalid value, this prompt, and all parameters depending on this prompt, are invalid. | | |

¹ Change causes phone to restart or reboot.

<exchange/>

You must set the connection parameters for the Microsoft Exchange application if you want users to be able to use the Calendaring feature. This feature is supported only on VVX 500 and 1500 phones and SpectraLink handsets.

Table 13-23: Microsoft Exchange Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| exchange.meeting.phonePattern | String | Null |
| The pattern used to identify phone numbers in meeting descriptions, where "x" denotes any digit and " " separates alternative patterns (for example, xxx-xxx-xxxx/604.xxx.xxxx). | | |
| exchange.meeting.reminderEnabled | 0 or 1 | 1 |
| If 0, meeting reminders are disabled. If 1, they are enabled. | | |
| exchange.server.url¹ | String | Null |
| The Microsoft Exchange server address. | | |

¹ Change causes phone to restart or reboot.

<feature/>

The feature parameter controls the activation or deactivation of a feature at run time.

Table 13-24: Feature Activation/Deactivation Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| feature.acdAgentAvailable.enabled¹ | 0 or 1 | 0 |
| If 0, the ACD agent available/unavailable feature is disabled. If 1, the feature is enabled. | | |
| feature.acdLoginLogout.enabled¹ | 0 or 1 | 0 |
| If 0, the ACD login/logout feature is disabled. If 1, the feature is enabled. | | |
| feature.acdPremiumUnavailability.enabled¹ | 0 or 1 | 0 |
| If 0, the premium ACD unavailability feature is disabled. If 1, premium ACD unavailability feature is enabled, and unavailability reason codes can be used (if the other ACD feature parameters are also be enabled). | | |
| feature.acdServiceControlUri.enabled¹ | 0 or 1 | 0 |
| If 0, the ACD service control URI feature is disabled. If 1, the feature is enabled. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| feature.audioVideoToggle.enabled | 0 or 1 | 0 |
| VVX 1500 only. If 0, the audio/video toggle feature is disabled. If 1, the feature is enabled. | | |
| feature.bluetooth.enabled | 0 or 1 | 1 |
| SpectralLink only. If 0, the Bluetooth headset feature is disabled. If 1, the feature is enabled. | | |
| feature.callCenterStatus.enabled | 0 or 1 | 0 |
| If 0, the Status Event Threshold capability is disabled. If 1, the Status Event Threshold capability is enabled. | | |
| feature.callList.enabled¹
All locally controlled call lists. | 0 or 1 | 1 |
| feature.callListMissed.enabled¹
The missed calls list. | 0 or 1 | 1 |
| feature.callListPlaced.enabled¹
The placed calls list. | 0 or 1 | 1 |
| feature.callListReceived.enabled¹
The received calls list. | 0 or 1 | 1 |
| If 0, the call list is disabled. If 1, the call list is enabled. To enable the Missed, Placed, or Received call lists, <code>feature.callList.enabled</code> must be enabled. Note: You cannot disable the call list feature on the SoundPoint IP 321/331/335. | | |
| feature.callPark.enabled¹ | 0 or 1 | 0 |
| If 0, the call park and call retrieve features are disabled. If 1, the features are enabled. | | |
| feature.callRecording.enabled¹ | 0 or 1 | 0 |
| VVX 1500 phones and SoundPoint IP phones with a USB port only. If 0, the call recording and playback feature is disabled. If 1, the feature is enabled. | | |
| feature.corporateDirectory.enabled | 0 or 1 | 0 |
| If 0, the corporate directory feature is disabled. If 1, the feature is enabled. | | |
| feature.directedCallPickup.enabled¹ | 0 or 1 | 0 |
| If 0, the directed call pickup feature is disabled. If 1, the feature is enabled. | | |
| feature.directory.enabled | 0 or 1 | 1 |
| If 0, the local contact directory is disabled. If 1, the directory is enabled. | | |
| feature.enhancedCallDisplay.enabled | 0 or 1 | 0 |
| If 0, the phone may display the protocol at the end of the called party identification (for example, <code>1234567 [SIP]</code>). If 1, the phone will display the number only (for example, <code>1234567</code>). | | |
| feature.enhancedFeatureKeys.enabled | 0 or 1 | 0 |
| If 0, the enhanced feature keys feature is disabled. If 1, the feature is enabled. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| feature.exchangeCalendar.enabled¹ | 0 or 1 | 0 |
| VVX 500 and 1500 and SpectraLink only. If 0, the calendaring feature is disabled. If 1, the feature is enabled. | | |
| feature.groupCallPickup.enabled¹ | 0 or 1 | 0 |
| If 0, the group call pickup feature is disabled. If 1, the feature is enabled. | | |
| feature.hoteling.enabled | 0 or 1 | 0 |
| If 0, Hoteling is disabled. If 1, Hoteling is enabled. | | |
| feature.lastCallReturn.enabled¹ | 0 or 1 | 0 |
| If 0, the last call return feature is disabled. If 1, the feature is enabled. | | |
| feature.messaging.enabled¹ | 0 or 1 | 0 |
| If 0, the instant messaging feature is disabled. If 1, the feature is enabled. | | |
| feature.nonVolatileRingerVolume.enabled | 0 or 1 | 1 |
| If 0, user changes to the ringer volume are reset to default when the phone reboots. If 1, user changes to the ringer volume are saved and maintained when the phone reboots. | | |
| feature.nWayConference.enabled | 0 or 1 | 0 |
| Always disabled on SoundPoint IP 321/331/335. Always enabled on VVX 500 and 1500.
If 0, the n-way conferencing managing feature is disabled and while three-way conferencing can exist, there is no manage conference page. If 1, n-way conferencing is enabled, conferences with the maximum number of parties are allowed, and the manage conference page is shown. | | |
| feature.pictureFrame.enabled¹ | 0 or 1 | 1 |
| VVX 500 and 1500 only. If 0, the digital picture frame feature is disabled. If 1, the digital picture frame feature is enabled. | | |
| feature.presence.enabled¹ | 0 or 1 | 0 |
| If 0, the presence feature — including buddy managements and user status — is disabled. If 1, the presence feature is enabled with the buddy and status options. | | |
| feature.ringDownload.enabled¹ | 0 or 1 | 1 |
| If 0, the phone will not download ringtones when it starts up. If 1, the phone will download ringtones when it starts up. | | |
| feature.urlDialing.enabled | 0 or 1 | 1 |
| If 0, URL/name dialing is not available. If 1, URL/name dialing is available from private lines. <i>Note:</i> If enabled, unknown callers will be identified on the display by their phone's IP address. | | |

¹ Change causes phone to restart or reboot.

These settings control the phone's ability to dynamically load an external font file during boot up. Loaded fonts can either overwrite pre-existing fonts embedded within the software (not recommended) or can extend the phone's font support for Unicode ranges not already embedded. The font file must be a Microsoft **.fnt** file format. The font file name must adhere to the following specific pattern:

Font filename:

<fontName>_<fontHeightInPixels>_<fontRange>.fnt

- <fontName> is a free string of characters that typically carries the meaning of the font. Examples are `fontFixedSize` for a fixed-size font, or `fontProportionalSize` for a proportional size font.
- <fontHeightInPixels> describes the font height in number of screen pixels.
- <fontRange> describes the Unicode range covered by this font. Since **.fnt** occurs in 256 character based blocks, the <fontRange> is `Uxx00_UxxFF` (**.fnt** file). For more information, see
- Setting the Phone Language.

Overwriting an Existing Font

If it is necessary to overwrite an existing font, use <fontName>_<fontHeightInPixels>:

SoundPoint IP 321, 331, and 335

| | |
|---------------------------------|---|
| <code>fontProp_10</code> | The font used for the idle display and default time display. |
| <code>fontPropSoftkey_10</code> | The font used for soft keys labels and menu titles. |
| <code>fontFixed7_10</code> | The font used for the status line, pop-up text, and the microbrowser. |

SoundPoint IP 550, 560, and 650

| | |
|--------------------------|--|
| <code>fontProp_12</code> | The font used for the audio progress bar and the microbrowser. |
| <code>fontProp_19</code> | The font used in the current implementation including for soft keys. |
| <code>fontProp_26</code> | The font used to display time (but not date). |
| <code>fontProp_mb</code> | This is a small font used for the CPU/Load/Net utilization graphs. |

SoundStation IP 5000 and 6000

| | |
|---------------------------------|--|
| <code>fontProp_10</code> | The font used for the idle display and the microbrowser. |
| <code>fontPropSoftkey_10</code> | This is a small font used for the CPU/Load/Net utilization graphs. |
| <code>fontProp_16</code> | The font used in the current implementation. |

If the values in <fontName>_<fontHeightInPixels> do not match any of the names above, then the downloaded font will be applied against all fonts defined in the phone, and you may lose the benefit of fonts being calibrated differently depending on their usage. For example, the font used to display the time on the SoundPoint IP 650 is larger than the one used to display the date, and if you overwrite this default font with a unique font, you lose this size aspect. For example:

- to overwrite the font used for SoundPoint IP 550 soft keys for ASCII, the name should be **fontPropSoftkey_10_U0000_U00FF.fnt**
- to add support for a new font that will be used everywhere and that is not currently supported. For example, for the Eastern/Central European Czech language, this is Unicode range 100-17F, the name could be **fontCzechIP500_10_U0100_U01FF.fnt** and **fontCzechIP600_19_U0100_U01FF.fnt**

Table 13-25: Font Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|----------------|
| font.delimiter | string up to 256 ASCII characters | Null |
| This parameter is not required. | | |
| font.x.name | fontName_height_Uxx00_UxxFF.fnt | Null |
| Defines the font file that will be loaded from the provisioning server during boot up. | | |

¹ Change causes phone to restart or reboot.

<hoteling/>

The Hoteling enhancement to ACD enables agents to use any available host phone by logging in with agent credentials. After logging in, agents have access to their guest profile and ACD settings on the host phone. The Hoteling enhancement is independent of the ACD feature, meaning agents can use Hoteling whether the Premium ACD feature is enabled or disabled.

Table 13-26: Hoteling Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| hoteling.reg | 1 – 34 | 1 |
| If ACD is disabled, the phone will use this line registration index for Hoteling. If ACD is enabled, this value must be the same as the ACD line registration index. | | |

<httpd/>

The phone contains a local Web Configuration Utility server for user and administrator features. This can be disabled for applications where it is not needed or where it poses a security threat. The Web server supports both basic and digest authentication. The authentication user name and password are not configurable for this release.

Table 13-27: HTTPD (Web Server) Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| httpd.enabled¹ | 0 or 1 | 1 |
| If 0, the HTTP server is disabled (the Web Configuration Utility will also be disabled). If 1, the server will be enabled. | | |
| httpd.cfg.enabled¹ | 0 or 1 | 1 |
| If 0, the Web Configuration Utility is disabled. If 1, the Web Configuration Utility is enabled. | | |
| httpd.cfg.port¹ | 1 to 65535 | 80 |
| Port is 80 for HTTP servers. Care should be taken when choosing an alternate port. | | |
| httpd.cfg.secureTunnelEnabled¹ | 0 or 1 | 1 |
| If 0, the Web does not use a secure tunnel. If 1, the server connects through a secure tunnel. | | |
| httpd.cfg.secureTunnelPort¹ | 1 to 65535 | 443 |
| The port to use for communications when the secure tunnel is used. | | |
| httpd.cfg.secureTunnelRequired¹ | 0 or 1 | 0 |
| If 0, communications to the Web server do not require a secure tunnel. If 1, communications do require a secure tunnel. | | |

¹ Change causes phone to restart or reboot.

<key/>

You can change the functions of your phone's keypad keys from the factory defaults, although this is typically not necessary. This process is also known as remapping. If you want to change the function of a key, you must specify the phone model and key to change, as well as the new function for the key. See Default Feature Key Layouts to find the key number for each key.



Caution: Key Remapping is Not Recommended

Polycom does not recommend remapping or changing the default functions of the keys on your phone.

Table 13-28: Key Parameters

| <i>Parameter</i> | <i>Permitted Values</i> |
|--|---|
| key.x.y.function.prim | A phone model string listed in Table 14-27 |
| The function for key y on phone model x. See <i>Table 14-27</i> for the x and y definitions. | |
| key.x.y.subPoint.prim | A key number listed in Table 14-27 |
| The sub-identifier for key functions with a secondary array identifier such as SpeedDial. | |

¹ Change causes phone to restart or reboot.

Table 13-29: Key Number on Polycom Phones

| <i>Phone Model (x in Table 14-26)</i> | <i>Key Numbers (y in Table 14-26)</i> |
|---------------------------------------|---------------------------------------|
| SPIP321, SPIP331, SPIP335 | 1 to 34 |
| SPIP450 | 1 to 35 |
| SPIP550, SPIP560 | 1 to 40 |
| SPIP650 | 1 to 42 |
| SSIP5000 | 1 to 32 |
| SSIP6000 | 1 to 29 |
| SSDUO | 1 to 35 |
| VVX 500 | 1 to 40 |
| VVX1500 | 1 to 42 |
| SL8440 | 1 to 28 |
| SL8450, 8452 | 1 to 29 |

The following table lists the functions that are available for the keys:

Table 13-30: Keypad Key Functions

| | | | | |
|-----------|----------|-----------|---------|----------|
| ArrowDown | Dialpad2 | Green | Menu | SoftKey3 |
| ArrowLeft | Dialpad3 | Handsfree | MicMute | SoftKey4 |

| | | | | |
|-------------|--------------|----------|----------|---------------|
| ArrowDown | Dialpad2 | Green | Menu | SoftKey3 |
| ArrowRight | Dialpad4 | Headset | MyStatus | SpeedDial |
| ArrowUp | Dialpad5 | Hold | Null | SpeedDialMenu |
| Back | Dialpad6 | Home | Offline | Talk |
| BarCode | Dialpad7 | Line1 | Red | Transfer |
| BuddyStatus | Dialpad8 | Line2 | Redial | Video |
| CallList | Dialpad9 | Line3 | Release | VolDown |
| Conference | DialpadStar | Line4 | Select | VolUp |
| Delete | DialPound | Line5 | Setup | |
| Dialpad0 | Directories | Line6 | SoftKey1 | |
| Dialpad1 | DoNotDisturb | Messages | SoftKey2 | |

<keypadLock/>

This parameter is supported on only SpectraLink handsets.

Table 13-31: Keypad Lock Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| keypadLock.enabled¹ | 0 or 1 | 1 |
| If 0, the keypad lock feature is disabled. If 1, the feature is enabled. | | |
| keypadLock.idleTimeout¹ | 0 to 65535 | 0 |
| The maximum time (in seconds) the handset can be idle before the keypad will lock. | | |

¹ Change causes phone to restart or reboot.

<lcl/>

The phone has a multilingual user interface. It supports both North American and international time and date formats as well as SoundStation Duo preferences.



Caution: Use a Multilingual XML Editor

Edit the language parameters using a multilingual XML editor. If you do not use an XML editor, some of the language labels in the configuration file, and in the language menu on the phone, will display incorrectly. To confirm whether your editor properly supports these characters, view the language parameter for languages such as Chinese, Japanese, Korean, Russian— for example `lcl.ml.lang.menu.1.label`.

This parameter definition also includes:

- `<ml/>`—The multilingual definition
- `<datetime/>`—The date and time definition
- [SoundStation Duo Localization Preferences](#)

<ml/>

The multilingual feature is based on string dictionary files downloaded from the provisioning server. These files are encoded in standalone XML format. Several eastern European and Asian languages are included with the distribution. Space for user-defined languages is available.

Table 13-32: Multilingual Parameters

| <i>Parameter</i> | <i>Permitted Values</i> |
|---|---|
| lcl.ml.lang | Null or an exact match for one of the label names stored in lcl.ml.lang.menu.x.label |
| If Null, the default internal language (US English) will be used, otherwise, the language to be used may be specified in the format of <code>lcl.ml.lang.menu.x.label</code> . For example, to get the phone to boot up in German, set this parameter to <code>Deutsch (de-de)</code> . | |
| lcl.ml.lang.charset¹ | string |
| The language character set. | |
| lcl.ml.lang.clock.x.24HourClock | 0 or 1 |
| If parameter present, overrides <code>lcl.datetime.time.24HourClock</code>
If 1, display time in 24-hour clock mode rather than am/pm. | |

| <i>Parameter</i> | <i>Permitted Values</i> |
|---|---|
| lcl.ml.lang.clock.x.dateTop | 0 or 1 |
| If parameter present, overrides <code>lcl.datetime.date.dateTop</code> .
If 1, display date above time, otherwise display time above date. | |
| lcl.ml.lang.clock.x.format | string which includes 'D', 'd' and 'M' and two optional commas |
| If parameter present, overrides <code>lcl.datetime.date.format</code> ;
D = day of week d = day M = month. Up to two commas may be included.
For example: D,dM = Thursday, 3 July or Md,D = July 3, Thursday
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" is illegal. | |
| lcl.ml.lang.clock.x.longFormat | 0 or 1 |
| If parameter present, overrides <code>lcl.datetime.date.longFormat</code> .
If 1, display the day and month in long format (Friday/November), otherwise use abbreviations (Fri/Nov). | |
| lcl.ml.lang.font.x¹ | string |
| The language font. | |
| lcl.ml.lang.list¹ | a comma-separated list |
| A list of the languages supported on the phones. Phone-specific parameters are defined for the SoundPoint IP 321/331/335 phones as they do not support Asian languages. | |
| lcl.ml.lang.menu.x | String in the format language_region |
| Dictionary file | |
| lcl.ml.lang.menu.x.label¹ | String in the format nativeLanguageName (abbreviation) |
| Phone language menu label | |
| The phone supports multiple languages. Dictionary files and labels must be sequential (for example, <code>lcl.ml.lang.menu.1</code> , <code>lcl.ml.lang.menu.2</code> , <code>lcl.ml.lang.menu.3</code> ... <code>lcl.ml.lang.menu.N</code>) The dictionary file cannot have caps, and the strings must exactly match a folder name of a dictionary file (you can find the names in the SoundPointIPLocalization folder of your software distribution). If you edit these parameters, you need to use a multilingual XML editor that supports Unicode, such as XML Notepad 2007.
For example, a dictionary file and label for German would be: <code>lcl.ml.lang.menu.8="German_Germany"</code>
<code>lcl.ml.lang.menu.8.label="Deutsch (de-de)"</code> | |

¹ Change causes phone to restart or reboot.

To add a new language:

- 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 `SoundPointIPLocalization` 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.

The basic character support includes the following Unicode character ranges.

Table 13-33: Unicode Ranges for Basic Character Support

| Name | Range |
|------------------------------------|-----------------|
| C0 Controls and Basic Latin | U+0000 - U+007F |
| C1 Controls and Latin-1 Supplement | U+0080 - U+00FF |
| Cyrillic (partial) | U+0400 - U+045F |

<datetime/>

The following parameters configure the date and time display on the phone.

Table 13-34: Date and Time Parameters

| Parameter | Permitted Values | Default |
|---|---|---------|
| lcl.datetime.date.dateTop | 0 or 1 | |
| If set to 1, display date above time else display time above date. | | |
| lcl.datetime.date.format | string which includes 'D', 'd' and 'M' and two optional commas | |
| Controls format of date string. D = day of week, d = day, M = month.
Up to two commas may be included.
For example: <code>D, dM = Thursday, 3 July</code> or <code>Md, D = July 3, Thursday</code>
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" is illegal. | | |
| lcl.datetime.date.longFormat | 0 or 1 | |
| If set to 1, display the day and month in long format (Friday/November), otherwise, use abbreviations (Fri/Nov). | | |

| Parameter | Permitted Values | Default |
|---|------------------|---------|
| lcl.datetime.time.24HourClock | 0 or 1 | |
| If set to 1, display time in 24-hour clock mode rather than a.m./p.m. | | |

SoundStation Duo Localization Preferences

The following table describes localization preferences that are specific to the SoundStation Duo phone.

Table 13-35: SoundStation Duo Localization Preferences

| Name | Permitted Values | Default |
|--|---|------------|
| lcl.flashTiming | 80, 100, 300, 600 (ms) | 600 |
| The length of time before a hook flash times-out (or the call disconnects). The flash duration is based on the country of origin that is specified for the phone. | | |
| lcl.callerId | On, Off, Removed | On |
| Caller ID displays a caller's phone number (and possibly a name), on the called party's phone. Specify whether caller ID is on, off, or removed. If caller ID is removed, the <i>Caller ID Type</i> menu item is removed from the phone's menu. | | |
| Note: <i>Caller ID is a subscription service. Check with your local telephone service provider to determine if this service is available in your area. Caller ID is not supported in Japan. If the phone is being used in Japan, choose the 'Removed' option.</i> | | |
| lcl.callerIdType | 1 (Bellcore)
2 (ETSI)
3 (British Telecom)
4 (DTMF)
5 (Off) | 1 |
| The caller ID standard to use for the phone. Note: <i>The British Telecom Caller ID standard is not supported with Polycom UC software 4.0.0B.</i> | | |
| lcl.dtmfLevel | -30 to 3 (dB) | -7 |
| The dual-tone multi-frequency (DTMF) level is the strength of the signal that is generated when you press a key on your phone. | | |
| lcl.dtmfTwist | 0 to 30 (dBV) | 20 |
| The difference between the high and low frequencies of the DTMF pair. | | |

| <i>Name</i> | <i>Permitted Values</i> | <i>Default</i> |
|-----------------|--|----------------|
| lcl.aidt | Auto (0), 2, 3, 4, 5, 6, 7,
Disabled (-1) (seconds) | 3 |

Automatic Initiation Delay Timing (AIDT) is the amount of time that the system waits for a dial tone before dialing a number during on-hook dialing.

Auto—Dial tone detected by software or 3 seconds, whichever comes first.

Disabled—Automatic off-hook dialing is disabled.

| | | |
|-----------------------------|--|---------------|
| lcl.pstnCountryIndex | Number, from 1 to 73
1-Argentina, 2-Australia,
3-Austria, 4-Bahrain, 5-
Belgium, 6-Brazil,
7-Bulgaria, 8-Canada, 9-Chile,
10-China, 11-Columbia, 12-
Croatia, 13-Europe (TBR21),
14-Cyprus,
15-Czech Republic,
16-Denmark, 17-Ecuador,
18-Egypt, 19-El Salvador,
20-Finland, 21-France,
22-Germany, 23-Greece, 24-
Guam, 25-Hong Kong, 26-
Hungary, 27-Iceland, 28-India,
29-Indonesia, 30-Ireland, 31-
Israel, 32-Italy, 33-Japan, 34-
Jordan, 35-Kazakhstan, 36-
Kuwait, 37-Latvia, 38-
Lebanon, 39-Luxembourg,
40-Macao, 41-Malaysia, 42-
Malta, 43-Mexico, 44-Morocco,
45-Netherlands, 46-New
Zealand, 47-Nigeria,
48-Norway, 49-Oman, 50-
Pakistan, 51-Peru,
52-Philippines, 53-Poland,
54-Portugal, 55-Romania,
56-Russia, 57-Saudi Arabia,
58-Singapore, 59-Slovakia,
60-Slovenia, 61-South Africa,
62-South Korea, 63-Spain,
64-Sweden, 65-Switzerland,
66-Syria, 67-Taiwan, 68-
Thailand, 69-UAE, 70-UK, 71-
USA, 72-Yemen | 71-USA |
|-----------------------------|--|---------------|

The country the phone operates in.

<license/>

This parameter's settings control aspects of the feature licensing system.

Table 13-36: Feature License Parameter

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| license.polling.time¹ | 00:00 – 23:59 | 02:00 |

The time (using the 24-hour clock) to check if the license has expired.

¹ Change causes phone to restart or reboot.



Note: Removing the Installed License

Once the license is installed on a phone, it cannot be removed.

<lineKey/>

The Flexible Line Key Assignment feature uses the <lineKey/> parameter.

Table 13-37: Line Key Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---------------------------------------|---|-------------------|
| lineKey.x.category¹ | unassigned, line, BLF, speedDial, presence | unassigned |

The line key category. Set the category to unassigned to leave a blank line key.

| | | |
|------------------------------------|------------------|----------|
| lineKey.x.index¹ | 0 to 9999 | 0 |
|------------------------------------|------------------|----------|

For lines, the index for line numbers. For speed dials, the speed dial index. For BLF or presence, 0. For unassigned, the value is ignored.

| | | |
|---|---------------|----------|
| lineKey.reassignment.enabled¹ | 0 or 1 | 0 |
|---|---------------|----------|

If 1, flexible line key assignment is enabled.

¹ Change causes phone to restart or reboot.

<loc/>

The values you enter for these Lync Server-only parameters will be used by E.911 services.

Table 13-38

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| locInfo.x.label | String | Null |
| Enter a label for your location. | | |
| locInfo.x.country | String | Null |
| Enter the country the phone is located in. | | |
| locInfo.x.A1 | String | Null |
| Enter the national subdivision the phone is located in, for example, a state or province. | | |
| locInfo.x.A3 | String | Null |
| Enter the city the phone is located in. | | |
| locInfo.x.PRD | String | Null |
| Enter the leading direction of the street location. | | |
| locInfo.x.RD | String | Null |
| The name of the road or street the phone is located on. | | |
| locInfo.x.STS | String | Null |
| Enter the suffix of the name used in locInfo.x.RD, for example, Street, Avenue. | | |
| locInfo.x.POD | String | Null |
| Enter the trailing street direction, for example SW. | | |
| locInfo.x.HNO | String | Null |
| Enter the street address number of the phone's location. | | |
| locInfo.x.HNS | String | Null |
| Enter a suffix for the street address used in locInfo.x.HNS, for example, ^A or ½. | | |
| locInfo.x.LOC | String | Null |
| Enter any additional information that identifies the location. | | |
| locInfo.x.NAM | String | Null |
| Enter a name for the location, for example, a business name, an occupant, a resident. | | |
| locInfo.x.PC | String | Null |
| Enter the postal code of the location. | | |

<log/>**Caution: Changing the Logging Parameters**

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

The event logging system supports the following classes of events:

Table 13-39: Logging Levels

| <i>Logging Level</i> | <i>Interpretation</i> |
|----------------------|---|
| 0 | Debug only |
| 1 | High detail class event |
| 2 | Moderate detail event class |
| 3 | Low detail event class |
| 4 | Minor error – graceful recovery |
| 5 | Major error – will eventually incapacitate the system |
| 6 | Fatal error |

Each event in the log contains the following fields separated by the | character:

- time or time/date stamp
- 1-5 character component identifier (such as “so”)
- event class
- cumulative log events missed due to excessive CPU load
- free form text - the event description

Three formats are available for the event timestamp:

Table 13-40: Event Timestamp Formats

| <i>Type</i> | <i>Example</i> |
|---|---|
| 0 - seconds.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 -- October 28, 17:16:42 |

Two types of logging are supported:

- `<level/> <change/>and<render/>`
- `<sched/>`

`<level/> <change/>and<render/>`

This configuration parameter is defined as follows:

Table 13-41: Logging Level, Change, and Render Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-----------------------------------|----------------|
| log.level.change.xxx | 0 to 6 | 4 |
| Control the logging detail level for individual components. These are the input filters into the internal memory-based log system. Possible values for xxx are acom, ares, app1, barcode, bluet, bdiag, brow, cap, cdp, cert, cfg, cipher, clink, clist, cmp, cmr, copy, curl, daa, dbs, dbuf, dhcpc, dis, dock, dot1x, dns, drvtbt, ec, efk, ethf, h323, hset, httpa, httpd, hw, ht, ib, key, ldap, lic, lldp, loc, log, mb, mobil, net, niche, oaip, oosp, osd, pcap, pcd, pdc, peer, pgui, pmt, pnetm, poll, pps, pres, pstn, ptt, push, pwrsv, rdisk, res, rtos, rtls, sec, sig, sip, slog, so, soem, srtp, sshc, ssps, style, sync, sys, ta, task, tls, trace, ttrs, usb, usbio, util, utilm, wdog, wifi, wlan, wmgr, and xmpp. | | |
| log.render.file | 0 or 1 | 1 |
| Set to 1. Polycom recommends that you do not change this value. | | |
| log.render.file.size | positive integer, 1 to 180 | 32 |
| Maximum size of flash memory for logs in Kbytes. When this size is about to be exceeded, the phone will upload all logs that have not yet been uploaded, and erase half of the logs on the phone. The administrator may use Web browser to read all logs on the phone. | | |
| log.render.file.upload.append | 0 or 1 | 1 |
| If set to 1, use append mode when uploading log files to server.
Note: HTTP and TFTP don't support append mode unless the server is set up for this. | | |
| log.render.file.upload.append.limitMode | delete, stop | delete |
| Behavior when server log file has reached its limit.
delete=delete file and start over stop=stop appending to file | | |
| log.render.file.upload.append.sizeLimit | positive integer | 512 |
| Maximum log file size that can be stored on provisioning server in Kbytes. | | |
| log.render.file.upload.period | positive integer | 172800 |
| Time in seconds between log file uploads to the provisioning server.
Note: The log file will not be uploaded if no new events have been logged since the last upload. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| log.render.level | 0 to 6 | 1 |
| <p>Specifies the lowest class of event that will be rendered to the log files. This is the output filter from the internal memory-based log system.</p> <p>The log.render.level maps to syslog severity as follows:</p> <p>0 -> SeverityDebug (7)
 1 -> SeverityDebug (7)
 2 -> SeverityInformational (6)
 3 -> SeverityInformational (6)
 4 -> SeverityError (3)
 5 -> SeverityCritical (2)
 6 -> SeverityEmergency (0) For more information, refer to Syslog Menu on page 64.</p> | | |
| log.render.realtime | 0 or 1 | 1 |
| <p>Set to 1. Polycom recommends that you do not change this value.</p> | | |
| log.render.stdout | 0 or 1 | 1 |
| <p>Set to 1. Polycom recommends that you do not change this value. Note that on SpectraLink handsets, the default value is 0.</p> | | |
| log.render.type | 0 to 2 | 2 |
| <p>Refer to Table 14-35: Event Timestamp Formats for timestamp type.</p> | | |

<sched/>

The phone can be configured to schedule certain advanced logging tasks on a periodic basis. These parameters should be set in 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.

Table 13-42: Logging Schedule Parameters

| <i>Parameter</i> | <i>Permitted Values</i> |
|--|-------------------------------------|
| log.sched.x.level | 0 to 5, default 3 |
| <p>Event class to assign to the log events generated by this command. This needs to be the same or higher than log.level.change.slog for these events to display in the log.</p> | |
| log.sched.x.name | alphanumeric string |
| <p>Name of an internal system command to be periodically executed. To be supplied by Polycom.</p> | |
| log.sched.x.period | positive integer, default 15 |
| <p>Seconds between each command execution. 0=run once</p> | |

| <i>Parameter</i> | <i>Permitted Values</i> |
|--|----------------------------------|
| log.sched.x.startDay | 0 to 7 |
| When startMode is <i>abs</i> , specifies the day of the week to start command execution. 1=Sun, 2=Mon, ..., 7=Sat | |
| log.sched.x.startMode | abs, rel |
| Start at an <i>absolute</i> time or <i>relative</i> to boot. | |
| log.sched.x.startTime | positive integer OR hh:mm |
| Seconds since boot when startMode is <i>rel</i> or the start time in 24-hour clock format when startMode is <i>abs</i> . | |

<mb/>

This parameter's settings control the home page, proxy and size limits to be used by the microbrowser and browser when it is selected to provide services. The microbrowser is supported on the SoundPoint IP 450, 550, 560, 601, and 650, and the SoundStation IP 6000 phones, and the Web browser is supported on the VVX 500 and 1500 phones and the SpectraLink handsets.

Table 13-43: Microbrowser and Web Browser Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|----------------|
| mb.idleDisplay.home | Null or any fully formed valid HTTP URL. Length up to 255 characters. | Null |
| The URL for the microbrowser home page that is shown on the idle display microbrowser Home page. For example: http://www.example.com/xhtml/frontpage. If Null, there is no idle display microbrowser. Note that the microbrowser idle display will displace the idle display indicator. | | |
| mb.idleDisplay.refresh | 0 or an integer > 5 | 0 |
| The time period in seconds that the microbrowser's idle display will refresh. If set to 0, the idle display microbrowser does not refresh. The minimum refresh period is 5 seconds (values from 1 to 4 are ignored, and 5 is used).
<i>Note:</i> If an HTTP Refresh header is detected, it will be respected, even if this parameter is set to 0. The refresh parameter will be respected only in the event that a refresh fails. Once a refresh is successful, the value in the HTTP refresh header, if available, will be used. | | |
| mb.main.autoBackKey¹ | 0 or 1 | 1 |
| If 0, the phone does not provide a Back soft key; all soft keys are created and controlled by the application. If 1, the phone automatically supplies a Back soft key in all main browser screens. The Back soft key will take the user back to the previous page in the browser history. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---|----------------------------------|
| mb.main.home | Any fully formed valid HTTP URL. Length up to 255 characters. | Null |
| The URL of the microbrowser's Home page. For example: <i>http://www.example.com/xhtml/frontpage/home</i> . If blank, the browser will notify the user that a blank home-page was used. | | |
| mb.main.idleTimeout | 0 - 600, seconds | 40 |
| The timeout, in seconds, for the interactive browser. If the interactive browser remains idle for the defined period of time, the phone returns to the idle browser. If 0, there is no timeout. | | |
| mb.main.statusbar | 0 or 1 | 0 |
| If 0, the status bar does not display. If 1, the status bar displays and status messages are shown. | | |
| mb.main.toolbar.autoHide.enabled² | 0 or 1 | 1 |
| If 0, the toolbar displays continually. If 1, the toolbar disappears if not selected. | | |
| mb.proxy | Null or domain name or IP address in the format <address>:<port> | Null. Default port = 8080 |
| The address of the HTTP proxy to be used by the microbrowser. If blank, normal unproxied HTTP is used by the microbrowser. | | |
| mb.ssawc.call.mode | active or passive | passive |
| Control the spontaneous display of Web content. If set to <i>passive</i> , Web content is displayed only when requested by the user. If set to <i>active</i> , Web content is displayed immediately. | | |
| mb.ssawc.enabled | 0 or 1 | 0 |
| If 0, spontaneous display of Web content is disabled. If 1, spontaneous Web content display is enabled. | | |

¹ Change causes phone to restart or reboot.

² For the SpectraLink 8400 Series handsets, the toolbar autohide is disabled by default.

<messaging/>

This parameter's setting control aspects of instant messaging on only the SpectraLink handsets.

Table 13-44: SpectraLink Instant Messaging Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| messaging.maxImMessages | 10 to 1000 | 1000 |
| The maximum number of instant messages allowed. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---------------------------------------|----------------|
| messaging.quickNotes.x | String of up to 128 characters | Null |
| Up to 10 (x =1 to 10) quick notes for use in instant messages | | |

<msg/>

Message-waiting indication is supported on a per-registration basis.

In the following table, x is the registration number. IP 321/331/335: x=1-2; IP 450: x=1-3; IP 550, 560: x=1-4; VVX 500:x=1-12; VVX 1500: x=1-6; IP 650; IP 5000, 6000, 7000: x=1.

Table 13-45: Message Waiting Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|---------------------|
| msg.bypassInstantMessage¹ | 0 or 1 | 0 |
| This parameter determines what is shown on the phone menu when you press the Messages key. If 0, the phone shows Message Center and Instant Messages. If 1, the phone bypasses these menus and goes directly to voicemail. This parameter applies only to supported phone models that have a Messages (MSG) key: SoundPoint IP (except IP 3xx models) and VVX 1500 business media phones. | | |
| msg.mwi.x.subscribe | 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 |
| If non-Null, the phone will send a SUBSCRIBE request to this contact after boot-up. | | |
| msg.mwi.x.callBackMode | contact, registration, disabled | registration |
| The message retrieval mode and notification for registration x. <i>contact</i> – a call is placed to the contact specified by <code>msg.mwi.x.callback</code> . <i>registration</i> – the registration places a call to itself (the phone calls itself). <i>disabled</i> – message retrieval and message notification are disabled. | | |
| msg.mwi.x.callBack | 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 |
| The contact to call when retrieving messages for this registration if <code>msg.mwi.x.callBackMode</code> is set to <i>contact</i> . | | |

¹ Change causes phone to restart or reboot.

<nat/>

These parameters define port and IP address changes used in NAT traversal. The port changes will change the port used by the phone, while the IP entry simply changes the IP advertised in the SIP signaling. This allows the use of simple NAT devices that can redirect traffic, but does not allow for port mapping. For example, port 5432 on the NAT device can be sent to port 5432 on an internal device, but not to port 1234.

Table 13-46: Network Access Translation

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-----------------------------------|----------------|
| nat.ip¹ | dotted- decimal IP address | Null |
| IP address to advertise within SIP signaling - should match the external IP address used by the NAT device. | | |
| nat.keepalive.interval | 0 to 3600 | 0 |
| The keep-alive interval in seconds. Sets the interval at which phones will 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 will not send out keep-alive messages.
The Microsoft Live Communications Server 2005 keep-alive feature will override this parameter. If you want to deploy phones behind a NAT and connect them to Live Communications Server, the keep-alive interval received from the Live Communications Server must be short enough to keep the NAT port open. Once the TCP connection is closed, the phones stop sending keep-alive packets. | | |
| nat.mediaPortStart¹ | 0 to 65440 | 0 |
| The initially allocated RTP port. Overrides the value set for <code>tcIpApp.port.rtp.mediaPortRangeStart</code> . | | |
| nat.signalPort¹ | 1024 to 65535 | 0 |
| The port used for SIP signaling. Overrides <code>voIpProt.local.port</code> . | | |

¹ Change causes phone to restart or reboot.

<np/>

This section shows you how to choose a default notification profile from four available types - Normal, Silent, Meeting, Custom1 - and shows you the parameters you can set for each type. Note that all phones use only the Normal profile type except the SpectraLink 84xx Series wireless handsets which can use any of the four types. Each profile is defined by an alert type and a ring type; there are 15 alert types and three ringing types.

For each alert type:

- You can select a tone pattern from the patterns defined in `se.pat.misc`. These patterns include: **custom1** to **custom10**, **instantMessaging**, **localHoldNotification**, **messageWaiting**, **misc1** to **misc9**, **negativeConfirm**, **positiveConfirm**, **remoteHoldNotification**, **silent**, and **welcome**. For information on customizing these parameters, refer to `se.pat.misc`.
- You can determine if the handset should vibrate for the alert, set the `vibrate` parameter to 0 to disable vibration or 1 to enable vibration.

For each ringer type:

- You can choose a tone pattern from the patterns defined in `se.pat.ringer`. These patterns include: **default**, **ringer1** to **ringer 24**, and **1** to **22**.
- You can also set the vibration type for the ringer. You can select **off**, **continuous**, **shortPulse**, or **longPulse**.

You can choose the default notification profile by configuring the parameter shown in the following table:

Table 13-47: Notification Profile Selection Parameter

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--------------------|---|----------------|
| np.selected | Normal, Silent, Meeting, Custom1 | Normal |

The initial profile that is selected when the phone powers on and active during operation. The user can override this default profile to set a new default profile that will be selected when the phone powers on the next time.

The configuration parameters for each profile type are described in the following tables:

- [Normal Profile Alert Parameters](#)
- [Silent Profile Alert Parameters](#)
- [Meeting Profile Alert Parameters](#)
- [Custom1 Profile Alert Parameters](#)

The table shown next illustrates the parameters you will need to configure to customize the Normal notification profile.

Table 13-48: Normal Profile Alert Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|------------------------|-------------------------|----------------|
| np.normal.label | String | Normal |

The name of the profile type.

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|--|
| np.normal.alert.barcodeBeep.tonePattern
np.normal.alert.barcodeBeep.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc2
0 |
| The tone pattern and vibration (1 to enable) for the alert played when a barcode is scanned. | | |
| np.normal.alert.docked.tonePattern
np.normal.alert.docked.vibration | Any tone (see
se.pat.misc)
0 or 1 | positiveConfirm
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the handset is docked. | | |
| np.normal.alert.undocked.tonePattern
np.normal.alert.undocked.vibration | Any tone (see
se.pat.misc)
0 or 1 | negativeConfirm
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the handset is undocked. | | |
| np.normal.alert.instantMessaging.tonePattern
np.normal.alert.instantMessaging.vibration | Any tone (see
se.pat.misc)
0 or 1 | instantMessage
0 |
| The tone pattern and vibration (1 to enable) for the instant message alert. | | |
| np.normal.alert.localHoldNotification.tonePattern
np.normal.alert.localHoldNotification.vibration | Any tone (see
se.pat.misc)
0 or 1 | localHoldNotification
0 |
| The tone pattern and vibration (1 to enable) for the local hold notification alert. | | |
| np.normal.alert.lossOfNetwork.tonePattern
np.normal.alert.lossOfNetwork.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc1
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the network is lost. | | |
| np.normal.alert.lowBattery.tonePattern
np.normal.alert.lowBattery.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc1
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the battery is low. | | |
| np.normal.alert.veryLowBattery.tonePattern
np.normal.alert.veryLowBattery.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc1
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the battery is very low. | | |
| np.normal.alert.messageWaiting.tonePattern
np.normal.alert.messageWaiting.vibration | Any tone (see
se.pat.misc)
0 or 1 | messageWaiting
0 |
| The tone pattern and vibration (1 to enable) for the alert played if there is a message waiting. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|------------------------------------|
| np.normal.alert.negativeConfirm.tonePattern
np.normal.alert.negativeConfirm.vibration | Any tone (see se.pat.misc)
0 or 1 | negativeConfirm
0 |
| The tone pattern and vibration (1 to enable) for the negative confirmation alert. | | |
| np.normal.alert.positiveConfirm.tonePattern
np.normal.alert.positiveConfirm.vibration | Any tone (see se.pat.misc)
0 or 1 | positiveConfirm
0 |
| The tone pattern and vibration (1 to enable) for the positive confirmation alert. | | |
| np.normal.alert.pttTransmit.tonePattern
np.normal.alert.pttTransmit.vibration | Any tone (see se.pat.misc)
0 or 1 | misc3
0 |
| The tone pattern and vibration (1 to enable) for the alert played if sending a push-to-talk page. | | |
| np.normal.alert.pttWait.tonePattern
np.normal.alert.pttWait.vibration | Any tone (see se.pat.misc)
0 or 1 | misc4
0 |
| The tone pattern and vibration (1 to enable) for the push-to-talk wait alert. | | |
| np.normal.alert.welcome.tonePattern
np.normal.alert.welcome.vibration | Any tone (see se.pat.misc)
0 or 1 | Welcome
0 |
| The tone pattern and vibration (1 to enable) for the alert played when the handset turns on. | | |
| np.normal.ringing.calls.tonePattern | A ringer (see se.pat.ringer) | default |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for normal calls. | | |
| np.normal.ringing.calls.vibration | off, continuous, shortPulse, longPulse | off |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for normal calls. | | |
| np.normal.ringing.oai1.tonePattern | A ringer (see se.pat.ringer) | ringer2 |
| The ringtone (see se.pat.ringer) for Open Application Interface (OAI) communications. | | |
| np.normal.ringing.oai1.vibration | off, continuous, shortPulse, longPulse | off |
| The vibration pattern for Open Application Interface (OAI) communications. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---|-------------------|
| np.normal.ringing.oai2.tonePattern | A ringer (see se.pat.ringer) | ringer2 |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for Open Application Interface (OAI) version 2.2 communications. | | |
| np.normal.ringing.oai2.vibration | off, continuous, shortPulse, longPulse | off |
| The vibration pattern for Open Application Interface (OAI) version 2.2 communications. | | |
| np.normal.ringing.privateLine.tonePattern | default, ringer1 to ringer24 | default |
| The ringtone (see se.pat.ringer) for a private line registered to Microsoft Lync Server 2010. | | |
| np.normal.ringing.privateLine.vibration | off, continuous, shortPulse, longPulse | shortPulse |
| The vibration pattern for a private line registered to Microsoft Lync Server 2010. | | |
| np.normal.ringing.toneVolume.handset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Handset and Normal profile is active. | | |
| np.normal.ringing.toneVolume.headset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Headset and Normal profile is active. | | |
| np.normal.ringing.toneVolume.chassis | -1000 to 1000 | 0 |
| The attribute is set (on adjusting ring volume) when ringing termination is Chassis and Normal profile is active. | | |
| np.normal.ringing.toneVolume.dock | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when phone is at the speakerphone dock and Normal profile is active. | | |
| np.normal.ringing.toneVolume.bluetoothHeadset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Bluetooth Headset and Normal profile is active. | | |
| np.normal.ringing.toneVolume.reserved | -1000 to 1000 | -21 |
| Not currently used. Reserved for future use. | | |
| np.normal.ringing.toneVolume.usbHeadset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is a USB headset and Normal profile is active. | | |

The table shown next illustrates the parameters you will need to configure to customize the Silent notification profile.

Table 13-49: Silent Profile Alert Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|---------------------|
| np.silent.label | String | Silent |
| The name of the profile type. | | |
| np.silent.alert.barcodeBeep.tonePattern
np.silent.alert.barcodeBeep.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the alert played when a barcode is scanned. | | |
| np.silent.alert.docked.tonePattern
np.silent.alert.docked.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the handset is docked. | | |
| np.silent.alert.undocked.tonePattern
np.silent.alert.undocked.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the handset is undocked. | | |
| np.silent.alert.instantMessaging.tonePattern
np.silent.alert.instantMessaging.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the instant message alert. | | |
| np.silent.alert.localHoldNotification.tonePattern
np.silent.alert.localHoldNotification.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the local hold notification alert. | | |
| np.silent.alert.lossOfNetwork.tonePattern
np.silent.alert.lossOfNetwork.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the network is lost. | | |
| np.silent.alert.lowBattery.tonePattern
np.silent.alert.lowBattery.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the battery is low. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|---------------------------|
| np.silent.alert.veryLowBattery.tonePattern
np.silent.alert.veryLowBattery.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the battery is very low. | | |
| np.silent.alert.messageWaiting.tonePattern
np.silent.alert.messageWaiting.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the alert played if there is a message waiting. | | |
| np.silent.alert.negativeConfirm.tonePattern
np.silent.alert.negativeConfirm.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the negative confirmation alert. | | |
| np.silent.alert.positiveConfirm.tonePattern
np.silent.alert.positiveConfirm.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the positive confirmation alert. | | |
| np.silent.alert.pttTransmit.tonePattern
np.silent.alert.pttTransmit.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the alert played if sending a push-to-talk page. | | |
| np.silent.alert.pttWait.tonePattern
np.silent.alert.pttWait.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the push-to-talk wait alert. | | |
| np.silent.alert.welcome.tonePattern
np.silent.alert.welcome.vibration | Any tone (see
se.pat.misc)
0 or 1 | silent
0 |
| The tone pattern and vibration (1 to enable) for the alert played when the handset turns on. | | |
| np.silent.ringing.calls.tonePattern | A ringer (see
se.pat.ringer) | ringer1 |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for normal calls. | | |
| np.silent.ringing.calls.vibration | off, continuous,
shortPulse, longPulse | off |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for normal calls. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---|-------------------|
| np.silent.ringing.oai1.tonePattern | A ringer (see se.pat.ringer) | ringer1 |
| The ringtone (see se.pat.ringer) for Open Application Interface (OAI) communications. | | |
| np.silent.ringing.oai1.vibration | off, continuous, shortPulse, longPulse | off |
| The vibration pattern for Open Application Interface (OAI) communications. | | |
| np.silent.ringing.oai2.tonePattern | A ringer (see se.pat.ringer) | ringer1 |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for Open Application Interface (OAI) version 2.2 communications. | | |
| np.silent.ringing.oai2.vibration | off, continuous, shortPulse, longPulse | off |
| The vibration pattern for Open Application Interface (OAI) version 2.2 communications. | | |
| np.silent.ringing.privateLine.tonePattern | default, ringer1 to ringer24 | ringer1 |
| The ringtone (see se.pat.ringer) for a private line registered to Microsoft Lync Server 2010. | | |
| np.silent.ringing.privateLine.vibration | off, continuous, shortPulse, longPulse | shortPulse |
| The vibration pattern for a private line registered to Microsoft Lync Server 2010. | | |
| np.silent.ringing.toneVolume.handset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Handset and Silent profile is active. | | |
| np.silent.ringing.toneVolume.headset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Headset and Silent profile is active. | | |
| np.silent.ringing.toneVolume.chassis | -1000 to 1000 | 0 |
| The attribute is set (on adjusting ring volume) when ringing termination is Chassis and Silent profile is active. | | |
| np.silent.ringing.toneVolume.dock | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when phone is at the speakerphone dock and Silent profile is active. | | |
| np.silent.ringing.toneVolume.bluetoothHeadset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Bluetooth Headset and Silent profile is active. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| np.silent.ringing.toneVolume.reserved | -1000 to 1000 | -21 |
| Not currently used. Reserved for future use. | | |
| np.silent.ringing.toneVolume.usbHeadset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is a USB headset and Silent profile is active. | | |

The table shown next illustrates the parameters you will need to configure to customize the Meeting notification profile.

Table 13-50: Meeting Profile Alert Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|------------------------------------|
| np.meeting.label | String | Meeting |
| The name of the profile type. | | |
| np.meeting.alert.barcodeBeep.tonePattern
np.meeting.alert.barcodeBeep.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc2
0 |
| The tone pattern and vibration (1 to enable) for the alert played when a barcode is scanned. | | |
| np.meeting.alert.docked.tonePattern
np.meeting.alert.docked.vibration | Any tone (see
se.pat.misc)
0 or 1 | positiveConfirm
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the handset is docked. | | |
| np.meeting.alert.undocked.tonePattern
np.meeting.alert.undocked.vibration | Any tone (see
se.pat.misc)
0 or 1 | negativeConfirm
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the handset is undocked. | | |
| np.meeting.alert.instantMessaging.tonePattern
np.meeting.alert.instantMessaging.vibration | Any tone (see
se.pat.misc)
0 or 1 | instantMessage
0 |
| The tone pattern and vibration (1 to enable) for the instant message alert. | | |
| np.meeting.alert.localHoldNotification.tonePattern
np.meeting.alert.localHoldNotification.vibration | Any tone (see
se.pat.misc)
0 or 1 | localHoldNotification
0 |
| The tone pattern and vibration (1 to enable) for the local hold notification alert. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|------------------------------------|
| np.meeting.alert.lossOfNetwork.tonePattern
np.meeting.alert.lossOfNetwork.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc1
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the network is lost. | | |
| np.meeting.alert.lowBattery.tonePattern
np.meeting.alert.lowBattery.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc1
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the battery is low. | | |
| np.meeting.alert.veryLowBattery.tonePattern
np.meeting.alert.veryLowBattery.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc1
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the battery is very low. | | |
| np.meeting.alert.messageWaiting.tonePattern
np.meeting.alert.messageWaiting.vibration | Any tone (see
se.pat.misc)
0 or 1 | messageWaiting
0 |
| The tone pattern and vibration (1 to enable) for the alert played if there is a message waiting. | | |
| np.meeting.alert.negativeConfirm.tonePattern
np.meeting.alert.negativeConfirm.vibration | Any tone (see
se.pat.misc)
0 or 1 | negativeConfirm
0 |
| The tone pattern and vibration (1 to enable) for the negative confirmation alert. | | |
| np.meeting.alert.positiveConfirm.tonePattern
np.meeting.alert.positiveConfirm.vibration | Any tone (see
se.pat.misc)
0 or 1 | positiveConfirm
0 |
| The tone pattern and vibration (1 to enable) for the positive confirmation alert. | | |
| np.meeting.alert.pttTransmit.tonePattern
np.meeting.alert.pttTransmit.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc3
0 |
| The tone pattern and vibration (1 to enable) for the alert played if sending a push-to-talk page. | | |
| np.meeting.alert.pttWait.tonePattern
np.meeting.alert.pttWait.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc4
0 |
| The tone pattern and vibration (1 to enable) for the push-to-talk wait alert. | | |
| np.meeting.alert.welcome.tonePattern
np.meeting.alert.welcome.vibration | Any tone (see
se.pat.misc)
0 or 1 | Welcome
0 |
| The tone pattern and vibration (1 to enable) for the alert played when the handset turns on. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---|-------------------|
| np.meeting.ringing.calls.tonePattern | A ringer (see se.pat.ringer) | ringer1 |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for normal calls. | | |
| np.meeting.ringing.calls.vibration | off, continuous, shortPulse, longPulse | continuous |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for normal calls. | | |
| np.meeting.ringing.oai1.tonePattern | A ringer (see se.pat.ringer) | ringer1 |
| The ringtone (see se.pat.ringer) for Open Application Interface (OAI) communications. | | |
| np.meeting.ringing.oai1.vibration | off, continuous, shortPulse, longPulse | continuous |
| The vibration pattern for Open Application Interface (OAI) communications. | | |
| np.meeting.ringing.oai2.tonePattern | A ringer (see se.pat.ringer) | ringer1 |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for Open Application Interface (OAI) version 2.2 communications. | | |
| np.meeting.ringing.oai2.vibration | off, continuous, shortPulse, longPulse | continuous |
| The vibration pattern for Open Application Interface (OAI) version 2.2 communications. | | |
| np.meeting.ringing.privateLine.tonePattern | default, ringer1 to ringer24 | ringer9 |
| The ringtone (see se.pat.ringer) for a private line registered to Microsoft Lync Server 2010. | | |
| np.meeting.ringing.privateLine.vibration | off, continuous, shortPulse, longPulse | shortPulse |
| The vibration pattern for a private line registered to Microsoft Lync Server 2010. | | |
| np.meeting.ringing.toneVolume.handset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Headset and Meeting profile is active. | | |
| np.meeting.ringing.toneVolume.headset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Headset and Meeting profile is active. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| np.meeting.ringing.toneVolume.chassis | -1000 to 1000 | 0 |
| The attribute is set (on adjusting ring volume) when ringing termination is Chassis and Meeting profile is active. | | |
| np.meeting.ringing.toneVolume.dock | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when phone is at the speakerphone dock and Meeting profile is active. | | |
| np.meeting.ringing.toneVolume.bluetoothHeadset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Bluetooth Headset and Meeting profile is active. | | |
| np.meeting.ringing.toneVolume.reserved | -1000 to 1000 | -21 |
| Not currently used. Reserved for future use. | | |
| np.meeting.ringing.toneVolume.usbHeadset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is a USB headset and Meeting profile is active. | | |

The table shown next illustrates the parameters you will need to configure to customize the Custom1 notification profile.

Table 13-51: Custom1 Profile Alert Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|------------------------------------|
| np.custom1.label | String | Custom1 |
| The name of the profile type. | | |
| np.custom1.alert.barcodeBeep.tonePattern
np.custom1.alert.barcodeBeep.vibration | Any tone (see se.pat.misc)
0 or 1 | misc2
0 |
| The tone pattern and vibration (1 to enable) for the alert played when a barcode is scanned. | | |
| np.custom1.alert.docked.tonePattern
np.custom1.alert.docked.vibration | Any tone (see se.pat.misc)
0 or 1 | postiveConfirm
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the handset is docked. | | |
| np.custom1.alert.undocked.tonePattern
np.custom1.alert.undocked.vibration | Any tone (see se.pat.misc)
0 or 1 | negativeConfirm
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the handset is undocked. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|--|
| np.custom1.alert.instantMessaging.tonePattern
np.custom1.alert.instantMessaging.vibration | Any tone (see
se.pat.misc)
0 or 1 | instantMessage
0 |
| The tone pattern and vibration (1 to enable) for the instant message alert. | | |
| np.custom1.alert.localHoldNotification.tonePattern
np.custom1.alert.localHoldNotification.vibration | Any tone (see
se.pat.misc)
0 or 1 | localHoldNotification
0 |
| The tone pattern and vibration (1 to enable) for the local hold notification alert. | | |
| np.custom1.alert.lossOfNetwork.tonePattern
np.custom1.alert.lossOfNetwork.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc1
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the network is lost. | | |
| np.custom1.alert.lowBattery.tonePattern
np.custom1.alert.lowBattery.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc1
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the battery is low. | | |
| np.custom1.alert.veryLowBattery.tonePattern
np.custom1.alert.veryLowBattery.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc1
0 |
| The tone pattern and vibration (1 to enable) for the alert played if the battery is very low. | | |
| np.custom1.alert.messageWaiting.tonePattern
np.custom1.alert.messageWaiting.vibration | Any tone (see
se.pat.misc)
0 or 1 | messageWaiting
0 |
| The tone pattern and vibration (1 to enable) for the alert played if there is a message waiting. | | |
| np.custom1.alert.negativeConfirm.tonePattern
np.custom1.alert.negativeConfirm.vibration | Any tone (see
se.pat.misc)
0 or 1 | negativeConfirm
0 |
| The tone pattern and vibration (1 to enable) for the negative confirmation alert. | | |
| np.custom1.alert.positiveConfirm.tonePattern
np.custom1.alert.positiveConfirm.vibration | Any tone (see
se.pat.misc)
0 or 1 | positiveConfirm
0 |
| The tone pattern and vibration (1 to enable) for the positive confirmation alert. | | |
| np.custom1.alert.pttTransmit.tonePattern
np.custom1.alert.pttTransmit.vibration | Any tone (see
se.pat.misc)
0 or 1 | misc3
0 |
| The tone pattern and vibration (1 to enable) for the alert played if sending a push-to-talk page. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|----------------------------|
| np.custom1.alert.pttWait.tonePattern
np.custom1.alert.pttWait.vibration | Any tone (see se.pat.misc)
0 or 1 | misc4
0 |
| The tone pattern and vibration (1 to enable) for the push-to-talk wait alert. | | |
| np.custom1.alert.welcome.tonePattern
np.custom1.alert.welcome.vibration | Any tone (see se.pat.misc)
0 or 1 | Welcome
0 |
| The tone pattern and vibration (1 to enable) for the alert played when the handset turns on. | | |
| np.custom1.ringing.calls.tonePattern | A ringer (see se.pat.ringer) | ringer2 |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for normal calls. | | |
| np.custom1.ringing.calls.vibration | off, continuous, shortPulse, longPulse | continuous |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for normal calls. | | |
| np.custom1.ringing.oai1.tonePattern | A ringer (see se.pat.ringer) | ringer2 |
| The ringtone (see se.pat.ringer) for Open Application Interface (OAI) communications. | | |
| np.custom1.ringing.oai1.vibration | off, continuous, shortPulse, longPulse | continuous |
| The vibration pattern for Open Application Interface (OAI) communications. | | |
| np.custom1.ringing.oai2.tonePattern | A ringer (see se.pat.ringer) | ringer2 |
| The ringtone (see se.pat.ringer) and vibration (1 to enable) for Open Application Interface (OAI) version 2.2 communications. | | |
| np.custom1.ringing.oai2.vibration | off, continuous, shortPulse, longPulse | continuous |
| The vibration pattern for Open Application Interface (OAI) version 2.2 communications. | | |
| np.custom1.ringing.privateLine.tonePattern | default, ringer1 to ringer24 | ringer9 |
| The ringtone (see se.pat.ringer) for a private line registered to Microsoft Lync Server 2010. | | |
| np.custom1.ringing.privateLine.vibration | off, continuous, shortPulse, longPulse | shortPulse |
| The vibration pattern for a private line registered to Microsoft Lync Server 2010. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| np.custom1.ringing.toneVolume.handset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Headset and Custom1 profile is active. | | |
| np.custom1.ringing.toneVolume.headset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Headset and Custom1 profile is active. | | |
| np.custom1.ringing.toneVolume.chassis | -1000 to 1000 | 0 |
| The attribute is set (on adjusting ring volume) when ringing termination is Chassis and Custom1 profile is active. | | |
| np.custom1.ringing.toneVolume.dock | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when phone is at the speakerphone dock and Custom1 profile is active. | | |
| np.custom1.ringing.toneVolume.bluetoothHeadset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is Bluetooth Headset and Custom1 profile is active. | | |
| np.custom1.ringing.toneVolume.reserved | -1000 to 1000 | -21 |
| Not currently used. Reserved for future use. | | |
| np.custom1.ringing.toneVolume.usbHeadset | -1000 to 1000 | -21 |
| The attribute is set (on adjusting ring volume) when ringing termination is a USB headset and Custom1 profile is active. | | |

<oai/>

The SpectraLink handsets support communications using the Open Application Interface (OAI). You can set the connection parameters using the table shown next:

Table 13-52: Open Application Interface (OAI) Parameters

| <i>Parameter</i> | <i>Permitted values</i> | <i>Default</i> |
|--------------------------------|-------------------------|----------------|
| oai.gateway.address | IP address | Null |
| The address of the OAI server. | | |

| <i>Parameter</i> | <i>Permitted values</i> | <i>Default</i> |
|-------------------|---|----------------|
| oai.userId | String of eight hexadecimal characters | Null |

The lower four bytes of the six-byte OAI handset identifier in the OAI gateway.

If the value is null or invalid, the handset identifies itself to the OAI gateway using the MAC address of the handset; otherwise, the upper two bytes are zero and the lower four bytes are as specified.

<phoneLock/>

The Enhanced Feature Key feature must be enabled if you want to use the **Lock** soft key.

Table 13-53: Phone Lock Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|------------------------------------|----------------|
| phoneLock.authorized.x.description
The name or description of an authorized number | String

string | |
| phoneLock.authorized.x.value
The number or address for an authorized contact | | |
| 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 or 1 | 0 |
| If 0, the microbrowser or browser is not displayed while the phone is locked. If 1, the microbrowser or browser is displayed while the phone is locked. | | |
| phoneLock.dndWhenLocked | 0 or 1 | 0 |
| If 0, the phone can receive calls while it is locked. If 1, the phone enters Do-Not-Disturb mode while it is locked. <i>Note:</i> The user can change this setting from the phone user interface. | | |
| phoneLock.enabled¹ | 0 or 1 | 0 |
| If 0, the phone lock feature is disabled. If 1, the phone lock feature is enabled. <i>Note:</i> To 'unlock' the phone remotely (in conjunction with deleting/modifying the overrides files), disable and re-enable this parameter. | | |
| phoneLock.idleTimeout | 0 to 65535 | 0 |
| The amount of time (in seconds) the phone can be idle before it automatically locks. If 0, automatic locking is disabled. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| phoneLock.lockState | 0 or 1 | 0 |
| <p>The value for this parameter indicates whether the phone is locked or unlocked and changes each time you lock or unlock the phone. If 0, the phone is unlocked. If 1, the phone is locked. Note that the phone stores and uploads the value each time it changes via the <code>MAC-phone.cfg</code>. You can set this parameter remotely using the Web Configuration Utility.</p> | | |
| phoneLock.powerUpUnlocked | 0 or 1 | 0 |
| <p>Use this parameter to override <code>phoneLock.lockState</code>. If 0, the phone retains the value in <code>phoneLock.lockState</code>. If 1, you can restart, reboot, or power cycle the phone to override the value for <code>phoneLock.lockState</code> in the <code>MAC-phone.cfg</code> and start the phone in an unlocked state. You can then lock or unlock the phone locally. Polycom recommends that you do not leave this parameter enabled.</p> | | |

¹ Change causes phone to restart or reboot.

<powerSaving/>

This parameter is supported on only the VVX 500 and 1500 phones.

The power saving feature automatically turns off the VVX 500 and 1500 phone's LCD display when it is not being used.

Table 13-54: Power Saving Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| powerSaving.enable | 0 or 1 | 1 |
| <p>If 0, the LCD power saving feature is disabled. If 1, the feature is enabled. <i>Note: The default value for the VVX 500 is 0.</i></p> | | |
| powerSaving.idleTimeout.offHours | 1 to 10 | 1 |
| <p>The number of minutes to wait while the phone is idle during off hours before activating power saving.</p> | | |
| powerSaving.idleTimeout.officeHours | 1 to 600 | 10 |
| <p>The number of minutes to wait while the phone is idle during office hours before activating power saving.</p> | | |
| powerSaving.idleTimeout.userInputExtension | 1 to 20 | 10 |
| <p>The minimum number of minutes to wait while the phone is idle — after the user uses the phone — before activating power saving.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| powerSaving.officeHours.duration.monday | 0 to 12 | 10 |
| powerSaving.officeHours.duration.tuesday | 0 to 12 | 10 |
| powerSaving.officeHours.duration.wednesday | 0 to 12 | 10 |
| powerSaving.officeHours.duration.thursday | 0 to 12 | 10 |
| powerSaving.officeHours.duration.friday | 0 to 12 | 10 |
| powerSaving.officeHours.duration.saturday | 0 to 12 | 0 |
| powerSaving.officeHours.duration.sunday | 0 to 12 | 0 |

The duration of the day's office hours.

| | | |
|--|----------------|----------|
| powerSaving.officeHours.startHour.xxx | 0 to 23 | 8 |
|--|----------------|----------|

The starting hour for the day's office hours, where xxx is one of `monday`, `tuesday`, `wednesday`, `thursday`, `friday`, `saturday`, and `sunday` (refer to `powerSaving.officeHours.duration` for an example).

| | | |
|--|----------------|----------|
| powerSaving.userDetectionSensitivity.offHours | 0 to 10 | 2 |
|--|----------------|----------|

The sensitivity of the algorithm used to detect the presence of the phone's user during off hours. 10 is the most sensitive. If set to 0, this feature is disabled.

The default value was chosen for good performance in a typical office environment and is biased for difficult detection during off hours.

| | | |
|---|----------------|----------|
| powerSaving.userDetectionSensitivity.officeHours | 0 to 10 | 7 |
|---|----------------|----------|

The sensitivity of the algorithm used to detect the presence of the phone's user during office hours. 10 is the most sensitive. If set to 0, this feature is disabled.

The default value was chosen for good performance in a typical office environment and is biased for easy detection during office hours.

<pres/>

The parameter `pres.reg` is the line number used to send SUBSCRIBE. If this parameter is missing, the phone will use the primary line to send SUBSCRIBE.

Table 13-55: Presence Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| pres.idleSoftkeys | 0 or 1 | 1 |
| If 0, the MyStat and Buddies presence idle soft keys do not display. If 1, the soft keys display. | | |
| pres.idleTimeout.offHours.enabled | 0 or 1 | 1 |
| If 0, the off hours idle timeout feature is disabled. If 1, the feature is enabled. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| pres.idleTimeout.offHours.period | 1 to 600 | 15 |
| The number of minutes to wait while the phone is idle during off hours before showing the Away presence status. | | |
| pres.idleTimeout.officeHours.enabled | 0 or 1 | 1 |
| If 0, the office hours idle timeout feature is disabled. If 1, the feature is enabled. | | |
| pres.idleTimeout.officeHours.period | 1 to 600 | 15 |
| The number of minutes to wait while the phone is idle during office hours before showing the Away presence status. | | |
| pres.reg | 1 to 34 | 1 |
| The valid line/registration number that is used for presence. This registration sends a SUBSCRIBE for presence. If the value is not a valid registration, this parameter is ignored. | | |

<prov/>

This parameter's settings control aspects of the phone's provisioning server system.

Table 13-56: Provisioning Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| prov.configUploadPath | string | Null |
| The directory - relative to the provisioning server - where the phone uploads the current configuration file when the user selects Upload Configuration. If set to Null, use the provisioning server directory. | | |
| prov.lineMap.cma.x¹ | 1 to 6 | 1 |
| Used to map the CMA H.323 line to a SIP line. Only x=1 is currently supported. | | |
| prov.login.automaticLogout | 0 to 46000 | 0 |
| The time (in minutes) before a non-default user is automatically logged out of the handset. If 0, the user is not automatically logged out. | | |
| prov.login.defaultPassword | String | Null |
| The login password for the default user. | | |
| prov.login.defaultOnly | 0 or 1 | 0 |
| If 1, the default user is the only user who can log in. If 0, other users can log in. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--------------------------|----------------|
| prov.login.defaultUser | String | Null |
| The username for the default user. If present, the user is automatically logged in when the phone boots up and logged in after another user logs out. | | |
| prov.login.enabled | 0 or 1 | 0 |
| If 0, the user profile feature is disabled. If 1, the user profile feature is enabled. | | |
| prov.login.lcCache.domain | 0 to 64 | Null |
| The user's sign-in domain name. | | |
| prov.login.lcCache.user | 0 to 64 | Null |
| The user's sign-in user name. | | |
| prov.login.localPassword | String | 123 |
| The password used to validate the user login. It is stored either as plain text or encrypted (an SHA1 hash). | | |
| prov.login.persistent | 0 or 1 | 0 |
| If 0, users are logged out if the handset reboots. If 1, users remain logged in when the phone reboots. | | |
| prov.login.required | 0 or 1 | 0 |
| If 1, a user must log in when the login feature is enabled. If 0, the user does not have to log in. | | |
| prov.loginCredPwdFlushed.enabled | 0 or 1 | 1 |
| If 1, when a user logs in or logs out, the login credential password is reset. If 0, the login credential password is not reset. | | |
| prov.polling.enabled | 0 or 1 | 0 |
| If 0, the provisioning server is not automatically polled for upgrades. If 1, the provisioning server is polled. | | |
| prov.polling.mode | abs, rel, random | abs |
| <p>The polling mode.</p> <p>abs The phone polls every day at the time specified by <code>prov.polling.time</code>.</p> <p>rel The phone polls after the number of seconds specified by <code>prov.polling.period</code>.</p> <p>random The phone polls at random between a starting time set in <code>prov.polling.time</code> and an end time set in <code>prov.polling.timeRandomEnd</code>. Note that if you set the polling period in <code>prov.polling.period</code> to a time greater than 86400 (one day) polling occurs on a random day between the start and end times based on the phone's MAC address.</p> | | |
| prov.polling.period | integer > 3600 | 86400 |
| The polling period in seconds. The polling period is rounded up to the nearest number of days in absolute mode. In relative mode, the polling period starts once the phone boots. In 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. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| prov.polling.time | hh:mm | 03:00 |
| The polling start time. Used in absolute and random modes. | | |
| prov.polling.timeRandomEnd | hh:mm | Null |
| The polling stop time. Only used in random mode. | | |
| prov.quickSetup.enabled | 0 or 1 | 0 |
| If 0, the quick setup feature is disabled. If 1, the quick setup feature is enabled. | | |
| prov.startupCheck.enabled | 0 or 1 | 1 |
| If 0, the phone is not provisioned at startup. If 1, the phone is provisioned at start up. All configuration files, licenses, and overrides are downloaded even if the software changes. (The previous behavior was to reboot as soon as the phone determined that software changed.) | | |

¹ Change causes phone to restart or reboot.

<pstn/>

The following table describes PSTN parameters that are specific to the SoundStation Duo phone.

Table 13-57: PSTN Parameters for the SoundStation Duo

| <i>Name</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|----------------|
| pstn.dateTimeFormat | 0 or 1 | 0 |
| <p>This parameter specifies whether or not to show the date on the idle screen and in call lists when the phone is running in PSTN mode and no SNTP server is specified or when Ethernet is down.</p> <p>If enabled, and the phone is running in PSTN mode and no SNTP server is specified or Ethernet is down, only the time will display on the idle screen and in call lists.</p> <p>If disabled, and the phone is running in PSTN mode and no SNTP server is specified or Ethernet is down, both the date (which may show the incorrect year) and time will display on the idle screen and in call lists.</p> <p>Note: By default, phones that operate in PSTN-only mode do not display the date and time unless:</p> <ul style="list-style-type: none"> • The date and time is set by an incoming call with a supported Caller ID standard. • The phone is connected to Ethernet and you turn on the date and time display. To turn on the time and date display, press the Menu key on the phone, and then select Settings > Basic > Preferences > Time and Date. | | |
| pstn.extension | Numerical string, up to a maximum of 32 numbers | Null |
| Your phone's telephone number. | | |

| <i>Name</i> | <i>Permitted Values</i> | <i>Default</i> |
|-----------------------------|-------------------------|----------------|
| pstn.useCallerIdTime | 0 or 1 | 1 |

This parameter specifies whether or not to include the time of the call when caller ID information displays on the phone.

<ptt/>

The PTT (push-to-talk) parameter is used to configure both Push-to-Talk and Group Paging features. Some of the parameters apply to both features while others apply to either PTT mode or page mode.

The parameters in the following table apply to PTT mode and page mode.

Table 13-58: Push-To-Talk and Group Paging Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-----------------------------|--------------------|
| ptt.address | multicast IP address | 224.0.1.116 |
| The multicast IP address to send page audio to and receive page audio from. | | |
| ptt.callWaiting.enable | 0 or 1 | 0 |
| If 0, incoming PTT sessions do not produce standard call waiting. If 1, incoming PTT sessions produce standard call waiting behavior on the active audio channel. | | |
| ptt.compatibilityMode | 0 or 1 | 1 |
| If 0, the PTT protocol behavior is not compatible with SpectraLink handset models 8020/8030 or older. If 1, all PTT protocol behavior is compatible with the older SpectraLink handsets, even if some configuration parameters are incompatible. For example, if this parameter is enabled and <code>ptt.codec</code> is set to G.722, the G.726QI codec will be used for outgoing PTT audio to maintain compatibility. | | |
| ptt.emergencyChannel.volume | -57 to 0 | -10 |
| The volume of emergency pages relative to the maximum speakerphone volume of the handset. 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. | | |
| ptt.port | 0 to 65535 | 5001 |
| The port to send audio to and receive audio from. | | |

The parameters in the following table apply to PTT mode push-to-talk only.

Table 13-59: Push-To-Talk Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--------------------------------|--|
| ptt.allowOffHookPages | 0 or 1 | 0 |
| If 0, PTT broadcasts will not play out on the handset during an active call — except for Priority and Emergency pages. If 1, PTT broadcasts will play out on the handset during an active call. | | |
| ptt.channel.x.available
Make the channel available to the user | 0 or 1 | 1 |
| ptt.channel.x.allowTransmit
Allow outgoing broadcasts on the channel | 0 or 1 | 1 |
| ptt.channel.x.label
The label to identify the channel | string | ch1: All, ch24: Priority, ch25: Emergency, others: Null |
| ptt.channel.x.subscribed
Subscribe the phone to the channel | 0 or 1 | ch1, 24, 25: 1, others: 0 |
| A push-to-talk channel x, where x= 1 to 25. The <code>label</code> is the name used to identify the channel during broadcasts. | | |
| If <code>available</code> is disabled (0), the user cannot access the channel or subscribe and the other channel parameters will be ignored. If enabled, the user can access the channel and choose to subscribe. | | |
| If <code>allowTransmit</code> is disabled (0), the user cannot sent PTT broadcasts on the channel. If enabled, the user may choose to send PTT broadcasts on the channel. | | |
| If <code>subscribed</code> is disabled, the phone will not be subscribed to the channel. If enabled, the phone will subscribe to the channel. | | |
| ptt.codec | G.711mu, G.726QI, G.722 | G.722 |
| The audio codec to use for outgoing PTT broadcasts. Incoming PTT audio will be decoded according to the codec specified in the incoming message. | | |
| ptt.defaultChannel | 1 to 25 | 1 |
| The PTT channel used to transmit an outgoing page if the user does not explicitly specify a channel. | | |
| ptt.displayName | string | Null |
| This display name is shown in the caller ID field of outgoing pages. If Null, the value from <code>reg.1.displayName</code> will be used. If the <code>reg.1</code> display name is also Null, the handset's MAC address will be used. | | |
| ptt.emergencyChannel | 1 to 25 | 25 |
| The channel assigned for emergency pages. | | |
| ptt.payloadSize | 10 to 80 | 20 |
| The audio payload size in milliseconds. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| ptt.priorityChannel | 1 to 25 | 24 |
| The channel assigned for priority pages. | | |
| ptt.pttMode.enable | 0 or 1 | 0 |
| If 0, push-to-talk is disabled. If 1, push-to-talk is enabled. | | |

The parameters in the following table apply to page mode group paging only.

Table 13-60: Group Paging Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|------------------------------------|----------------|
| ptt.pageMode.allowOffHookPages | 0 or 1 | 0 |
| If 0, group pages will not play out on the handset during an active call — except for Priority and Emergency pages. If 1, group pages will play out on the handset during an active call. | | |
| ptt.pageMode.codec | G.711Mu, G.726QI, or G.722 | G.722 |
| The audio codec to use for outgoing group pages. Incoming pages will be decoded according to the codec specified in the incoming message. | | |
| ptt.pageMode.defaultGroup | 1 to 25 | 1 |
| The paging group used to transmit an outgoing page if the user does not explicitly specify a group. | | |
| ptt.pageMode.displayName | up to 64 octet UTF-8 string | PTT |
| This display name is shown in the caller ID field of outgoing group pages. If Null, the value from <code>reg.1.displayName</code> will be used. If the <code>reg.1</code> display name is also Null, the handset's MAC address will be used. | | |
| ptt.pageMode.emergencyGroup | 1 to 25 | 25 |
| The paging group to use for emergency pages. | | |
| ptt.pageMode.enable | 0 or 1 | 0 |
| If 0, group paging is disabled. If 1, group paging is enabled. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---|--|
| ptt.pageMode.group.x.available
Make the group available to the user | 0 or 1 | 1 |
| ptt.pageMode.group.x.allowTransmit
Allow outgoing announcements to the group | 0 or 1 | 1 |
| ptt.pageMode.group.x.label
The label to identify the group | string | ch24: Priority,
ch25: Emergency,
others: Null |
| ptt.pageMode.group.x.subscribed
Subscribe the phone to the group | 0 or 1 | ch1, 24, 25: 1, others: 0 |
| <p>A page mode group x, where x= 1 to 25. The <code>label</code> is the name used to identify the group during pages. If <code>available</code> is disabled (0), the user cannot access the group or subscribe and the other page mode group parameters will be ignored. If enabled, the user can access the group and choose to subscribe. If <code>allowTransmit</code> is disabled (0), the user cannot send outgoing pages to the group. If enabled, the user may send outgoing pages. If <code>subscribed</code> is disabled, the phone will not be subscribed to the group. If enabled, the phone will subscribe to the group.</p> | | |
| ptt.pageMode.payloadSize | 10, 20, ..., 80
milliseconds | 20 |
| <p>The page mode audio payload size.</p> | | |
| ptt.pageMode.priorityGroup | 1 to 25 | 24 |
| <p>The paging group to use for priority pages.</p> | | |
| ptt.pageMode.transmit.timeout.continuation | 0 to 65535 | 60 |
| <p>The time (in seconds) to add to the initial timeout (<code>ptt.pageMode.transmit.timeout.initial</code>) for terminating page announcements. If this value is non-zero, an Extend soft key will display on the phone. Pressing the Extend soft key continues the initial timeout for the time specified by this parameter. If 0, announcements cannot be extended.</p> | | |
| ptt.pageMode.transmit.timeout.initial | 0 to 65535 | 0 |
| <p>The number of seconds to wait before automatically terminating an outgoing page announcement. If 0, page announcements will not automatically terminate.</p> | | |

<qbc/>

This parameter sets the connection parameters for the Quick Barcode Connector application on the SpectraLink handsets.

Table 13-61: Quick Barcode Connector (QBC) Application Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--------------------------------|----------------------------|
| qbc.connect.ipAddress.hostname | IP address or hostname | Null |
| The IP address or hostname of the computer running the Quick Barcode Connector application. Used in <code>single</code> endpoint mode only. | | |
| qbc.connect.passphrase | String | BcmaTest Password 1 |
| The barcode scanner connector passphrase. This value must match the passphrase in the QBC PC application even if <code>bcma.encrypted.enabled</code> is set to 0. The minimum length is 4 characters. | | |
| qbc.connection.port¹ | 0 to 65535 | 14394 |
| The Quick Barcode Connector application port number. | | |
| qbc.encrypted.enabled | 0 or 1 | 0 |
| If 0, scanned data is not encrypted. If 1, scanned data is encrypted. | | |
| qbc.inactivity.timeout | 30000 to 300000 | 60000 |
| The amount of time (in milliseconds) to wait before disconnecting the barcode scanner due to inactivity. | | |
| qbc.operating.mode | disabled, single, multi | multi |
| The Quick Barcode Connector application operating mode. If <code>disabled</code> the QBC application is disabled. If <code>single</code> , the application uses single endpoint mode (transfer to only one computer, one-to-one). If <code>multi</code> , the application uses multi endpoint mode (transfer to many computers, one-to-many). | | |

¹ Do not change unless directed by Polycom Customer Support

<qos/>

These parameters control the Quality of Service (QoS) options:

- The 802.1p/Q `user_priority` field RTP, call control, and other packets
- The “type of service” field RTP and call control packets

Table 13-62: Quality of Service (Type-of-Service) Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---|----------------|
| qos.ethernet.callControl.user_priority¹ | 0 to 7 | 5 |
| User-priority used for call control packets. | | |
| qos.ethernet.other.user_priority¹ | 0 to 7 | 2 |
| User-priority used for packets that do not have a per-protocol setting. | | |
| qos.ethernet.rtp.user_priority¹ | 0 to 7 | 5 |
| Choose the priority of voice Real-Time Protocol (RTP) packets. The default priority level is 5. | | |
| qos.ethernet.rtp.video.user_priority¹ | 0 to 7 | 5 |
| User-priority used for Video RTP packets. | | |
| qos.ip.callControl.dscp¹ | 0 to 63 or EF or any of AF11,AF12, AF13,AF21, AF22,AF23, AF31,AF32, AF33,AF41, AF42,AF43 | Null |
| Specify the DSCP of packets. If the value is not null, this parameter will override the other <code>qos.ip.callControl.*</code> parameters. The default value is Null, so the other <code>qos.ip.callControl.*</code> parameters will be used if no value is entered. | | |
| qos.ip.callControl.max_reliability¹ | 0 or 1 | 0 |
| qos.ip.callControl.max_throughput¹ | 0 or 1 | 0 |
| qos.ip.callControl.min_cost¹ | 0 or 1 | 0 |
| qos.ip.callControl.min_delay¹ | 0 or 1 | 1 |
| qos.ip.callControl.precedence¹ | 0 -7 | 5 |
| Set the bits in the IP ToS field of the IP header used for call control. Specify whether or not to set the max reliability bit, the max throughput bit, the min cost bit, the min delay bit, and the precedence bits. If 0, the bit in the IP ToS field of the IP header is not set. If 1, the bit is set. | | |
| qos.ip.rtp.dscp¹ | 0 to 63 or EF or any of AF11,AF12, AF13,AF21, AF22,AF23, AF31,AF32, AF33,AF41, AF42,AF43 | Null |
| Specify the DSCP of packets. If the value is not null, this parameter will override the other <code>qos.ip.rtp.*</code> parameters. The default value is Null, so the other <code>quality.ip.rtp.*</code> parameters will be used. | | |
| qos.ip.rtp.max_reliability¹ | 0 or 1 | 0 |
| qos.ip.rtp.max_throughput¹ | 0 or 1 | 1 |
| qos.ip.rtp.min_cost¹ | 0 or 1 | 0 |
| qos.ip.rtp.min_delay¹ | 0 or 1 | 1 |
| qos.ip.rtp.precedence¹ | 0 -7 | 5 |
| Set the bits in the IP ToS field of the IP header used for RTP. Specify whether or not to set the max reliability bit, the max throughput bit, the min cost bit, the min delay bit, and the precedence bit. If 0, the bit in the IP ToS field of the IP header is not set. If 1, the bit is set. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---|----------------|
| qos.ip.rtp.video.dscp¹ | 0 to 63 or EF or any of AF11,AF12, AF13,AF21, AF22,AF23, AF31,AF32, AF33,AF41, AF42,AF43 | Null |
| Allows the DSCP of packets to be specified. If the value is non-null, this parameter will override the other qos.ip.rtp.video.*parameters. The default value is Null, so the other qos.ip.rtp.video.* parameters will be used. | | |
| qos.ip.rtp.video.max_reliability¹ | 0 or 1 | 0 |
| qos.ip.rtp.video.max_throughput¹ | 0 or 1 | 1 |
| qos.ip.rtp.video.min_cost¹ | 0 or 1 | 0 |
| qos.ip.rtp.video.min_delay¹ | 0 or 1 | 1 |
| qos.ip.rtp.video.precedence¹ | 0-7 | 5 |
| Set the bits in the IP ToS field of the IP header used for RTP video. Specify whether or not to set the max reliability bit, the max throughput bit, the min cost bit, the min delay bit, and the precedence bit. If 0, the bit in the IP ToS field of the IP header is not set. If 1, the bit is set. | | |

¹ Change causes phone to restart or reboot.

<reg/>

SoundPoint IP 321/331/335 support a maximum of two unique registrations, SoundPoint IP 450 supports three, the SoundPoint IP 550 and 560 supports four, and SoundPoint IP 650, and the Polycom VVX 500 and 1500 support six. Up to three SoundPoint IP Expansion Modules can be added to a single host SoundPoint IP 650 phone to increase the total number of registrations to 34. Each registration can optionally be associated with a private array of servers for completely segregated signaling. The SoundStation IP 5000 and 6000, and the SoundStation Duo support a single registration. The SpectraLink handsets support six registrations.

In the following tables, x is the registration number. IP 321/331/335: x=1-2; IP 450: x=1-3; IP 550, 560: x=1-4; VVX 500: x=1-12; VVX 1500: x=1-6; IP 650: x=1-34; IP 5000, IP 6000, Duo: x=1; SL8400: x=1-6.

[Table 14-](#) and [Table 14-](#) show the Registration Parameters and the Server Registration Parameters:

Table 13-63: Registration Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| reg.x.acd-login-logout | 0 or 1 | 0 |
| reg.x.acd-agent-available | 0 or 1 | 0 |
| If both ACD login/logout and agent available are set to 1 for registration x, the ACD feature will be enabled for that registration. | | |
| reg.x.address | string address | Null |
| The user part (for example, 1002) or the user and the host part (for example, 1002@polycom.com) of the registration SIP URI or the H.323 ID/extension. | | |
| reg.x.applyServerDigitMapLocally | 0 or 1 | 0 |
| If 1 and <code>reg.x.server.y.specialInterop</code> is set to <code>lync2010</code> , the phone uses the dialplan from the Microsoft Lync Server. Any dialed number will apply the dial plan locally.
If 0, the dialplan from the Microsoft Lync Server is not used. | | |
| reg.x.auth.domain | string | Null |
| The domain of the authorization server that is used to check the user names and passwords. | | |
| reg.x.auth.optimizedInFailover | 0 or 1 | 0 |
| The destination of the first new SIP request when failover occurs. If 0, the SIP request is sent to the server with the highest priority in the server list. If 1, the SIP request is sent to the server which sent the proxy authentication request. | | |
| reg.x.auth.password | string | Null |
| The password to be used for authentication challenges for this registration. If the password is non-Null, it will override the password entered into the Authentication submenu on the Settings menu of the phone. | | |
| reg.x.auth.userId | string | Null |
| User ID to be used for authentication challenges for this registration. If the User ID is non-Null, it will override the user parameter entered into the Authentication submenu on the Settings menu of the phone. | | |
| reg.x.auth.useLoginCredentials | 0 or 1 | 0 |
| If 0, login credentials are not used for authentication to the server on registration x. If 1, login credentials are used for authentication to the server. <i>Note:</i> This must be set to 1 for instant messaging on the SpectraLink handsets. | | |
| reg.x.bargeInEnabled | 0 or 1 | 0 |
| If 0, barge-in is disabled for line x. If 1, barge-in is enabled (remote users of shared call appearances can interrupt or <i>barge in to</i> active calls). | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-----------------------------|---|
| reg.x.callsPerLineKey¹ | 1-4, 1-8, 1-24 | 24 (for IP 450, 550, 560, and 650, VVX, and SpectraLink phones)
8 (for all other phones) |
| 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 parameter overrides <code>call.callsPerLineKey</code> . | | |
| reg.x.csta | 0 or 1 | 0 |
| If 0, the uaCSTA (User Agent Computer Supported Telecommunications Applications) feature is disabled. If 1, uaCSTA is enabled (overrides the global parameter <code>voIpProt.SIP.csta</code>). | | |
| reg.x.dialPlanName | String | Null |
| If <code>reg.x.server.y.specialInterop</code> is set to <code>lync2010</code> , the dialplan name from the Microsoft Lync Server is stored here. Each registration has its own name for this dialplan. <i>Note:</i> Do not change this parameter if set by Microsoft Lync. | | |
| reg.x.displayName | UTF-8 encoded string | Null |
| The display name used in SIP signaling and/or the H.323 alias used as the default caller ID. | | |
| reg.x.filterReflectedBlDialogs | 0 or 1 | 1 |
| If 0, bridged line appearance NOTIFY messages (dialog state change) will not be ignored. If 1, the messages will be ignored. | | |
| reg.x.fwd.busy.contact | string | Null |
| The forward-to contact for calls forwarded due to busy status. If Null, the contact specified by <code>divert.x.contact</code> will be used. | | |
| reg.x.fwd.busy.status | 0 or 1 | 0 |
| If 0, incoming calls that receive a busy signal will not be forwarded. If 1, busy calls are forwarded to the contact specified by <code>reg.x.fwd.busy.contact</code> . | | |
| reg.x.fwd.noanswer.contact | string | Null |
| The forward-to contact used for calls forwarded due to no answer. If Null, the contact specified by <code>divert.x.contact</code> will be used. | | |
| reg.x.fwd.noanswer.ringCount | 0 to 65535 | 0 |
| The number of seconds the phone should ring for before the call is forwarded because of no answer. <i>Note:</i> The maximum value accepted by some call servers is 20. | | |
| reg.x.fwd.noanswer.status | 0 or 1 | 0 |
| If 0, calls are not forwarded if there is no answer. If 1, calls are forwarded to the contact specified by <code>reg.x.noanswer.contact</code> after ringing for the length of time specified by <code>reg.x.fwd.noanswer.ringCount</code> . | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|----------------|
| reg.x.ice.turn.callAdmissionControl.enabled | 0 or 1 | 0 |
| If 0, call admission control is disabled. If 1, call admission control is enabled for calls using the Microsoft Lync 2010 Server. | | |
| reg.x.label | UTF-8 encoded string | Null |
| The text label that displays next to the line key for registration x. If Null, the user part of <code>reg.x.address</code> is used. | | |
| reg.x.lcs | 0 or 1 | 0 |
| If 0, the Microsoft Live Communications Server (LSC) is not supported for registration x. If 1, LSC is supported. | | |
| reg.x.lineKeys | 1 to max | 1 |
| 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. To find out the maximum number for your phone, see Table 7-4: Flexible Call Appearances . | | |
| reg.x.lisdisclaimer | string, 0 to 256 characters | Null |
| 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." This parameter is set by in-band provisioning when the phone is registered to Microsoft Lync Server 2010. | | |
| reg.x.lync.autoProvisionCertLocation | 0 to 6 | 6 |
| If 0, the certificate download is disabled. If non-0, the certificate corresponding to the index of the appropriate <code>sec.TLS.customCaCert.X</code> is downloaded. | | |
| reg.x.musicOnHold.uri | a SIP URI | Null |
| A URI that provides the media stream to play for the remote party on hold. If present and not Null, this parameter overrides <code>voIpProt.SIP.musicOnHold.uri</code> . | | |
| reg.x.outboundProxy.address | dotted-decimal IP address or hostname | Null |
| The IP address or hostname of the SIP server to which the phone sends all requests. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---|--------------------|
| reg.x.outboundProxy.failOver.failBack.mode | newRequests
DNSTTL
registration
duration | newRequests |
| <p>The mode for failover failback (overrides <code>reg.x.server.y.failOver.failBack.mode</code>).</p> <p><code>newRequests</code> all new requests are forwarded first to the primary server regardless of the last used server.</p> <p><code>DNSTTL</code> 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.</p> <p><code>registration</code> the phone tries the primary server again when the registration renewal signaling begins.</p> <p><code>duration</code> the phone tries the primary server again after the time specified by <code>reg.x.outboundProxy.failOver.failBack.timeout</code> expires.</p> | | |
| reg.x.outboundProxy.failOver.failBack.timeout | 0, 60 to 65535 | 3600 |
| <p>The time to wait (in seconds) before failback occurs (overrides <code>reg.x.server.y.failOver.failBack.timeout</code>). 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. If 0, the phone will not fail-back until a fail-over event occurs with the current server.</p> | | |
| reg.x.outboundProxy.failOver.failRegistrationOn | 0 or 1 | 0 |
| <p>When set to 1, and the <code>reRegisterOn</code> parameter is enabled, the phone will silently invalidate an existing registration (if it exists), at the point of failing over. When set to 0, and the <code>reRegisterOn</code> 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.</p> <p>Note that <code>reg.x.outboundProxy.failOver.RegisterOn</code> must be enabled.</p> | | |
| reg.x.outboundProxy.failOver.onlySignalWithRegistered | 0 or 1 | 1 |
| <p>When set to 1, and the <code>reRegisterOn</code> and <code>failRegistrationOn</code> 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 will end. No SIP messages will be sent to the unregistered server. When set to 0, and the <code>reRegisterOn</code> and <code>failRegistrationOn</code> parameters are enabled, 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).</p> | | |
| reg.x.outboundProxy.failOver.reRegisterOn | 0 or 1 | 0 |
| <p>This parameters overrides <code>reg.x.server.y.failOver.failBack.RegisterOn</code>. When set to 1, the phone will attempt 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 will proceed with the secondary server. When set to 0, 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.</p> | | |
| reg.x.outboundProxy.port | 1 to 65535 | 0 |
| <p>The port of the SIP server to which the phone sends all requests.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|-----------------|
| reg.x.outboundProxy.transport | DNSnaptr, TCPpreferred, UDPOnly, TLS, TCPOnly | DNSnaptr |
| <p>The transport method the phone uses to communicate with the SIP server.</p> <p>Null or DNSnaptr – if <code>reg.x.outboundProxy.address</code> is a hostname and <code>reg.x.outboundProxy.port</code> is 0 or Null, do NAPTR then SRV look-ups to try to discover the transport, ports and servers, as per RFC 3263. If <code>reg.x.outboundProxy.address</code> is an IP address, or a port is given, then UDP is used.</p> <p>TCPpreferred – TCP is the preferred transport, UDP is used if TCP fails.</p> <p>UDPOnly – only UDP will be used.</p> <p>TLS – if TLS fails, transport fails. Leave port field empty (will default to 5061) or set to 5061.</p> <p>TCPOnly – only TCP will be used.</p> | | |
| reg.x.protocol.H323 | 0 or 1 | 0 |
| <p>VVX 1500 only. If 0, H.323 signaling is not enabled for registration x. If 1, H.323 signaling is enabled.</p> | | |
| reg.x.protocol.SIP | 0 or 1 | 1 |
| <p>VVX 1500 only. If 0, SIP signaling is not enabled for this registration. If 1, SIP signaling is enabled.</p> | | |
| reg.x.proxyRequire | string | Null |
| <p>The string that needs to be entered in the <i>Proxy-Require</i> header. If Null, no <i>Proxy-Require</i> will be sent.</p> | | |
| reg.x.ringType | default, ringer1 to ringer24 | ringer2 |
| <p>The ringer to be used for calls received by this registration. The default is the first non-silent ringer.</p> | | |
| reg.x.ringType.privateLine | default, ringer1 to ringer24 | default |
| <p>The ringer to be used for calls received by a private line connected to Microsoft Lync Server 2010.</p> | | |
| reg.x.serverAutoDiscovery | 0 or 1 | 1 |
| <p>Determines whether or not to discover the server address automatically. This parameter is used with Microsoft Lync Server 2010.</p> | | |
| reg.x.serverFeatureControl.cf¹ | 0 or 1 | 0 |
| <p>If 0, server-based call forwarding is not enabled, this is the old behavior. If 1, server based call forwarding is enabled. This parameter overrides <code>voIpProt.SIP.serverFeatureControl.cf</code>.</p> | | |
| reg.x.serverFeatureControl.dnd¹ | 0 or 1 | 0 |
| <p>If 0, server-based do-not-disturb (DND) is not enabled. If 1, server-based DND is enabled and the call server has control of DND. This parameter overrides <code>voIpProt.SIP.serverFeatureControl.dnd</code>.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|--------------------------------|
| reg.x.serverFeatureControl.localProcessing.cf | 0 or 1 | 1 |
| If 0 and <code>reg.x.serverFeatureControl.cf</code> is set to 1, the phone will not perform local Call Forward behavior. If set to 1, the phone will perform local Call Forward behavior on all calls received. This parameter overrides <code>voIpProt.SIP.serverFeatureControl.localProcessing.cf</code> . | | |
| reg.x.serverFeatureControl.localProcessing.dnd | 0 or 1 | 1 |
| If 0 and <code>reg.x.serverFeatureControl.dnd</code> is set to 1, the phone will not perform local DND call behavior. If set to 1, the phone will perform local DND call behavior on all calls received. This parameter overrides <code>voIpProt.SIP.serverFeatureControl.localProcessing.dnd</code> . | | |
| reg.x.serverFeatureControl.signalingMethod | string | serviceMsForwardContact |
| Controls the method used to perform call forwarding requests to the server. | | |
| reg.x.server.y.registerRetry.maxTimeout | | 180 seconds |
| Set the maximum period of time in seconds that you want the phone to try registering with the server. | | |
| reg.x.srtp.enable¹ | 0 or 1 | 1 |
| If 0, the registration always declines SRTP offers. If 1, the registration accepts SRTP offers. | | |
| reg.x.srtp.offer¹ | 0 or 1 | 0 |
| If 1, the registration includes a secure media stream description along with the usual non-secure media description in the SDP of a SIP INVITE. This parameter applies to the registration initiating (offering) a phone call. If 0, no secure media stream is included in SDP of a SIP invite. | | |
| reg.x.srtp.require¹ | 0 or 1 | 0 |
| If 0, secure media streams are not required. If 1, the registration is only allowed to use secure media streams. Any offered SIP INVITES must include a secure media description in the SDP or the call will be rejected. For outgoing calls, only a secure media stream description is included in the SDP of the SIP INVITE, meaning that the non-secure media description is not included. If this parameter set to 1, <code>reg.x.srtp.offer</code> will also be set to 1, regardless of the value in the configuration file. | | |
| reg.x.srtp.simplifiedBestEffort | 0 or 1 | 0 |
| If 0, no SRTP is supported. If 1, negotiation of SRTP compliant with Microsoft Session Description Protocol Version 2.0 Extensions is supported. This parameter overrides <code>sec.srtp.simplifiedBestEffort</code> . | | |
| reg.x.strictLineSeize | 0 or 1 | 0 |
| If 1, the phone is forced to wait for 200 OK on registration x when receiving a TRYING notify. If 0, the old behavior is used. This parameter overrides <code>voIpProt.SIP.strictLineSeize</code> for registration x. | | |
| reg.x.tcpFastFailover | 0 or 1 | 0 |
| If 1, failover occurs based on the values of <code>reg.x.server.y.retryMaxCount</code> and <code>voIpProt.server.x.retryTimeOut</code> . If 0, the old behavior is used. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--------------------------|----------------|
| reg.x.telephony | 0 or 1 | 1 |
| If 0, telephony calls are not enabled on this registration (use this value if the registration is used with Microsoft Office Communications Server 2007 R2 or Microsoft Lync 2010. If 1, telephony calls are enabled on this registration. | | |
| reg.x.thirdPartyName | string address | Null |
| This field must match the <code>reg.x.address</code> value of the registration which makes up the part of a bridged line appearance (BLA). It must be Null in all other cases. | | |
| reg.x.type | private or shared | private |
| If set to private, use standard call signaling. If set to shared, augment call signaling with call state subscriptions and notifications and use access control for outgoing calls. | | |
| reg.x.useCompleteUriForRetrieve | 0 or 1 | 1 |
| This parameters overrides <code>voipPort.SIP.useCompleteUriForRetrieve</code> . If set to 1, the target URI in BLF signaling will use the complete address as provided in the xml dialog document. If set to 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. | | |

¹ Change causes phone to restart or reboot.

You can list multiple registration servers for fault tolerance. In the following table, you can list 4 servers by using `y=1` to 4. If the `reg.x.server.y.address` is not null, all of the parameters in the following table will override the parameters specified in `voIpProt.server.*`. The server registration parameters are listed in the following table:

Table 13-64: Registration Server Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|----------------|
| reg.x.server.H323.y.address | dotted-decimal IP address or hostname | Null |
| Address of the H.323 gatekeeper. | | |
| reg.x.server.H323.y.port | 0 to 65535 | 0 |
| Port to be used for H.323 signaling. If set to Null, 1719 (H.323 RAS signaling) is used. | | |
| reg.x.server.H323.y.expires | positive integer | 3600 |
| Desired registration period. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---|--------------------|
| reg.x.server.y.address | dotted-decimal IP address or hostname | Null |
| <p>The IP address or host name of a SIP server that accepts registrations. If not Null, all of the parameters in this table will override the parameters specified in <code>voIpProt.server.*</code>. <i>Notes:</i> If this parameter is set, it will take precedence even if the DHCP server is available. If this registration is used for Microsoft Office Communications Server 2007 R2 on SpectraLink handsets, this parameter must be in the form <code>OCShostname.OSCdomain_name</code>.</p> | | |
| reg.x.server.y.expires | positive integer, minimum 10 | 3600 |
| <p>The phone's requested registration period in seconds. <i>Note:</i> The period negotiated with the server may be different. The phone will attempt to re-register at the beginning of the overlap period. For example, if <code>expires="300"</code> and <code>overlap="5"</code>, the phone will re-register after 295 seconds (300–5).</p> | | |
| reg.x.server.y.expires.lineSeize | 0 to 65535 | 30 |
| <p>Requested line-seize subscription period.</p> | | |
| reg.x.server.y.expires.overlap | 5 to 65535 | 60 |
| <p>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.</p> | | |
| reg.x.server.y.failOver.failBack.mode | newRequests DNSTTL registration duration | newRequests |
| <p>The mode for failover failback (this parameter overrides <code>voIpProt.server.x.failOver.failBack.mode</code>):</p> <ul style="list-style-type: none"> <code>newRequests</code> – all new requests are forwarded first to the primary server regardless of the last used server. <code>DNSTTL</code> – 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. <code>registration</code> – the phone tries the primary server again when the registration renewal signaling begins. <code>duration</code> – the phone tries the primary server again after the time specified by <code>reg.x.server.y.failOver.failBack.timeout</code>. | | |
| reg.x.server.y.failOver.failBack.timeout | 0, 60 to 65535 | 3600 |
| <p>The time to wait (in seconds) before failback occurs (overrides <code>voIpProt.server.x.failOver.failBack.timeout</code>). 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. If 0, the phone will not fail-back until a fail-over event occurs with the current server.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| reg.x.server.y.failOver.failRegistrationOn | 0 or 1 | 0 |
| 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. 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. | | |
| reg.x.server.y.failOver.onlySignalWithRegistered | 0 or 1 | 1 |
| 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 will end. No SIP messages will be sent to the unregistered server. When set to 0, and the reRegisterOn and failRegistrationOn parameters are enabled, 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). | | |
| reg.x.server.y.failOver.reRegisterOn | 0 or 1 | 0 |
| This parameter overrides the <code>voIpProt.server.x.failOver.reRegisterOn</code> . When set to 1, the phone will attempt 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 will proceed with the secondary server. When set to 0, 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. | | |
| reg.x.server.y.lcs | 0 or 1 | 0 |
| If 0, the Microsoft Live Communications Server (LSC) is not supported. If 1, LCS is supported for registration x. | | |
| reg.x.server.y.useOutboundProxy | 0 or 1 | 1 |
| Specify whether or not to use the outbound proxy specified in <code>reg.x.outboundProxy.address</code> for server x. This parameter overrides <code>voIpProt.server.x.useOutboundProxy</code> for registration x. | | |
| reg.x.server.y.port | 0, 1 to 65535 | Null |
| The port of the sip server that specifies registrations. If 0, the port used depends on <code>reg.x.server.y.transport</code> . | | |
| reg.x.server.y.register | 0 or 1 | 1 |
| If 0, calls can be routed to an outbound proxy without registration. See <code>voIpProt.server.x.register</code> . For more information, see Technical Bulletin 5844: SIP Server Fallback Enhancements on Polycom Phones . | | |
| reg.x.server.y.registerRetry.baseTimeOut | 10 - 120 | 60 |
| The base time period to wait before a registration retry. Used in conjunction with <code>reg.x.server.y.registerRetry.maxTimeOut</code> to determine how long to wait. The algorithm is defined in RFC 5626. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|-----------------|
| reg.x.server.y.registerRetry.maxTimeout | 60 - 1800 | 60 |
| <p>The maximum time period to wait before a registration retry. Used in conjunction with <code>reg.x.server.y.registerRetry.baseTimeout</code> to determine how long to wait. The algorithm is defined in RFC 5626.</p> | | |
| reg.x.server.y.retryMaxCount | 0 to 20 | 3 |
| <p>If set to 0, 3 is used. The number of retries that will be attempted before moving to the next available server.</p> | | |
| reg.x.server.y.retryTimeout | 0 to 65535 | 0 |
| <p>The amount of time (in milliseconds) to wait between retries. If 0, use standard RFC 3261 signaling retry behavior.</p> | | |
| reg.x.server.y.specialInterop | standard,
ocs2007r2,
lcs2005,
lync2010 | standard |
| <p>Specify if this registration should support Microsoft Office Communications Server 2007 R2 (<code>ocs2007r2</code>), Microsoft Live Communications Server 2005 (<code>lcs2005</code>), or Microsoft Lync 2010 (<code>lync2010</code>). <i>Note:</i> To use instant messaging on SpectraLink handsets, set this parameter to <code>ocs2007r2</code>.</p> | | |
| reg.x.server.y.transport | DNSnaptr,
TCPpreferred,
UDPOnly, TLS,
TCPOnly | DNSnaptr |
| <p>The transport method the phone uses to communicate with the SIP server.</p> <p>Null or <code>DNSnaptr</code> - if <code>reg.x.server.y.address</code> is a hostname and <code>reg.x.server.y.port</code> is 0 or Null, do NAPTR then SRV look-ups to try to discover the transport, ports and servers, as per RFC 3263. If <code>reg.x.server.y.address</code> is an IP address, or a port is given, then UDP is used.</p> <p><code>TCPpreferred</code> - TCP is the preferred transport; UDP is used if TCP fails.</p> <p><code>UDPOnly</code> - only UDP will be used.</p> <p><code>TLS</code> - if TLS fails, transport fails. Leave port field empty (will default to 5061) or set to 5061.</p> <p><code>TCPOnly</code> - only TCP will be used.</p> | | |

<request/>

These settings control the phone's behavior when a request for restart or reconfiguration is received.

Table 13-65: Configuration Request Parameter

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---------------------------------------|-------------------------|----------------|
| request.delay.type¹ | audio, call | call |

Specify when the phone should process a request for a restart or reconfiguration. If set to `audio`, the request will be executed once there is no active audio on the phone — regardless of the call state. If set to `call`, the request should be executed once there are no calls —in any state — on the phone.

¹ Change causes phone to restart or reboot.

<roaming_buddies/>

This parameter is used in conjunction with Microsoft Live Communications Server 2005 on most Polycom phones, Microsoft Office Communications Server 2007 R2 on the SpectraLink handsets, and Microsoft Lync on most Polycom phones.

Table 13-66: Roaming Buddies Parameter

| <i>Parameter</i> | <i>Permitted Value</i> | <i>Default</i> |
|----------------------------|------------------------|----------------|
| roaming_buddies.reg | 1 to 34 | Null |

The index of the registration which has roaming buddies support enabled. If Null, the roaming buddies feature is disabled. **Note:** This parameter must be set if the call server is Microsoft Live Communications Server 2005, Microsoft Office Communications Server 2007 R2, or Microsoft Lync.

<roaming_privacy/>

This parameter is used in conjunction with Microsoft Live Communications Server 2005 on most Polycom phones, Microsoft Office Communications Server 2007 R2 on the SpectraLink handsets, and Microsoft Lync on most Polycom phones.

Table 13-67: Roaming Privacy Parameters

| <i>Parameter</i> | <i>Permitted Value</i> | <i>Default</i> |
|----------------------------|------------------------|----------------|
| roaming_privacy.reg | 1 to 34 | Null |

Specify the index of the registration/line that has roaming privacy support enabled. If Null, roaming privacy is disabled.

<saf/>

The phone uses built-in wave files for some sound effects. The built-in wave files can be replaced with files downloaded from the provisioning server or from the Internet. However, these are stored in volatile memory so the files will need to remain accessible should the phone need to be rebooted. Files will be truncated to a maximum size of 300 kilobytes.

The following sampled audio WAVE (.wav) file formats are supported:

- mono 8 kHz G.711 u-Law
- G.711 A-Law
- L16/16000 (16-bit, 16 kHz sampling rate, mono)
- L16/32000 (16-bit, 32 kHz sampling rate, mono)
- L16/48000 (16-bit, 48 kHz sampling rate, mono)



Note: WAV Audio File Format Support

The L16/32000 and L16/49000 formats are supported on the VVX 1500 and SoundStation IP 5000 phones.

In the following table, x is the sampled audio file number.

Table 13-68: Sampled Audio File Parameter

| Parameter | Permitted Values | Default |
|--------------|--|---------|
| saf.x | Null or valid path name or an RFC 1738-compliant URL to a HTTP, FTP, or TFTP wave file resource. | |
| | <p>If Null, the phone will use a built-in file.</p> <p>If set to a path name, the phone will attempt to download this file at boot time from the provisioning server.</p> <p>If set to a URL, the phone will attempt to download this file at boot time from the Internet.</p> <p>Note: A TFTP URL is expected to be in the format: <code>tftp://<host>/[pathname]<filename></code>, for example: <code>tftp://somehost.example.com/sounds/example.wav</code>.</p> <p>Note: See the above wave file format restrictions.</p> | |

The following table defines the default usage of the sampled audio files with the phone:

Table 13-69: Default Sample Audio File Usage

| Sampled Audio File Number | Default Use (Pattern Reference) |
|---------------------------|--|
| 1 | Ringer 12 (<code>se.pat.misc.welcome</code>) |

| <i>Sampled Audio File Number</i> | <i>Default Use (Pattern Reference)</i> |
|----------------------------------|---|
| 2 | Ringer 13 (<code>se.pat.ringer.ringer15</code>) |
| 3 | Ringer 14 (<code>se.pat.ringer.ringer16</code>) |
| 4 | Ringer 15 (<code>se.pat.ringer.ringer17</code>) |
| 5 | Ringer 16 (<code>se.pat.ringer.ringer18</code>) |
| 6 | Ringer 17 (<code>se.pat.ringer.ringer19</code>) |
| 7 | Ringer 18 (<code>se.pat.ringer.ringer20</code>) |
| 8 | Ringer 19 (<code>se.pat.ringer.ringer21</code>) |
| 9 | Ringer 20 (<code>se.pat.ringer.ringer22</code>) |
| 10 | Ringer 21 (<code>se.pat.ringer.ringer23</code>) |
| 11 | Ringer 22 (<code>se.pat.ringer.ringer24</code>) |
| 12 to 24 | Not Used |



Note: Setting the Welcome Sound for SIP 3.1 and later

In SIP 3.1, the SoundPoint IP welcome sound was removed from `saf.1`. If you want the welcome sound to play when the phone reboots or restarts, set `saf.1` to `SoundPointIPWelcome.wav`.

<se/>

The phone uses both synthesized (based on the chord-sets, see <chord/>) and sampled audio sound effects. Sound effects are defined by patterns: rudimentary sequences of chord-sets, silence periods, and wave files.

Table 13-70: Sound Effect Parameter

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---------------------------------------|-------------------------|----------------|
| se.appLocalEnabled¹ | 0 or 1 | 1 |

If set to 1, local user interface sound effects such as confirmation/error tones, will be enabled.

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|----------------|
| se.destination | chassis, headset, handset, active | 1 |
| <p>The transducer or audio device that plays sound effects and alerts. Choose from the <code>chassis</code> (speakerphone), <code>headset</code> (if connected), <code>handset</code>, or the <code>active</code> destination. If <code>active</code>, alerts will play from the destination that is currently in use. For example, if you are in a call on the handset, a new incoming call will ring on the handset.</p> | | |
| se.stutterOnVoiceMail | 0 or 1 | 1 |
| <p>If set to 1, a stuttered dial tone is used in place of a normal dial tone to indicate that one or more voicemail messages are waiting at the message center.</p> | | |

This parameter also includes:

- `<pat/>`
- `<rt/>`

`<pat/>`

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 following instructions:

Table 13-71: Sound Effects Pattern Types

| <i>Instruction</i> | <i>Meaning</i> |
|--|--|
| sampled (n) | Play sampled audio file <i>n</i> |
| <p>Example:</p> <pre>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)</pre> | |
| chord (n, d) | Play chord set <i>n</i> (<i>d</i> is optional and allows the chord set ON duration to be overridden to <i>d</i> milliseconds) |
| <p>Example:</p> <pre>se.pat.callProg.busyTone.inst.2.type = "chord" (chord set instruction type) se.pat.callProg.busyTone.inst.2.value = "busyTone" (specifies sampled audio file <i>busyTone</i>) se.pat.callProg.busyTone.inst.2.param = "2000" (override ON duration of chord set to 2000 milliseconds)</pre> | |
| silence (d) | Play silence for <i>d</i> milliseconds (Rx audio is not muted) |
| <p>Example:</p> <pre>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)</pre> | |

| <i>Instruction</i> | <i>Meaning</i> |
|---|--|
| branch (n) | Advance n instructions and execute that instruction (n must be negative and must not branch beyond the first instruction) |
| <p>Example:</p> <pre>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)</pre> | |

In the following table, *x* is the pattern name, *y* is the instruction number. Both *x* and *y* need to be sequential. There are three categories *cat* of sound effect patterns: *callProg* (Call Progress Patterns), *ringer* (Ringer Patterns) and *misc* (Miscellaneous Patterns).

Table 13-72: Sound Effects Pattern Parameters

| <i>Parameter</i> | <i>Permitted Values</i> |
|--|--|
| se.pat.cat.x.name | UTF-8 encoded string |
| Sound effects name, where <i>cat</i> is <i>callProg</i> , <i>ringer</i> , or <i>misc</i> . | |
| se.pat.cat.x.inst.y.type | sampled, chord, silence, branch |
| Type of sound effect, where <i>cat</i> is <i>callProg</i> , <i>ringer</i> , or <i>misc</i> . | |
| se.pat.cat.x.inst.y.value | String |
| The instruction: <i>sampled</i> – sampled audio file number, <i>chord</i> – type of sound effect, <i>silence</i> – silence duration in ms, <i>branch</i> – number of instructions to advance. <i>cat</i> is <i>callProg</i> , <i>ringer</i> , or <i>misc</i> . | |

Call Progress Patterns

The following table shows the call progress pattern names and their descriptions:

Table 13-73: Call Progress Tone Pattern Names

| <i>Call Progress Pattern Name</i> | <i>Description</i> |
|-----------------------------------|--------------------------------------|
| alerting | Alerting |
| bargeln | Barge-in tone |
| busyTone | Busy tone |
| callWaiting | Call waiting tone |
| callWaitingLong | Call waiting tone long (distinctive) |

| <i>Call Progress Pattern Name</i> | <i>Description</i> |
|-----------------------------------|--------------------------------|
| confirmation | Confirmation tone |
| dialTone | Dial tone |
| howler | Howler tone (off-hook warning) |
| intercom | Intercom announcement tone |
| msgWaiting | Message waiting tone |
| precedenceCallWaiting | Precedence call waiting tone |
| precedenceRingback | Precedence ringback tone |
| preemption | Preemption tone |
| precedence | Precedence tone |
| recWarning | Record warning |
| reorder | Reorder tone |
| ringback | Ringback tone |
| secondaryDialTone | Secondary dial tone |
| stutter | Stuttered dial tone |

Ringer Patterns

The following table shows the ring pattern names and their default descriptions:

Table 13-74: Ringtone Pattern Names

| <i>Parameter Name</i> | <i>Ringtone Name</i> | <i>Description</i> |
|-----------------------|----------------------|----------------------------------|
| ringer1 | Silent Ring | Silent ring |
| ringer2 | Low Trill | Long single A3 Db3 major warble |
| ringer3 | Low Double Trill | Short double A3 Db3 major warble |
| ringer4 | Medium Trill | Long single C3 E3 major warble |
| ringer5 | Medium Double Trill | Short double C3 E3 major warble |
| ringer6 | High Trill | Long single warble 1 |
| ringer7 | High Double Trill | Short double warble 1 |

| <i>Parameter Name</i> | <i>Ringtone Name</i> | <i>Description</i> |
|-----------------------|----------------------|--|
| ringer8 | Highest Trill | Long single Gb3 A4 major warble |
| ringer9 | Highest Double Trill | Short double Gb3 A4 major warble |
| ringer10 | Beeble | Short double E3 major |
| ringer11 | Triplet | Short triple C3 E3 G3 major ramp |
| ringer12 | Ringback-style | Short double ringback |
| ringer13 | Low Trill Precedence | Long single A3 Db3 major warble Precedence |
| ringer14 | Ring Splash | Splash |
| ringer15 | Ring16 | Sampled audio file 1 |
| ringer16 | Ring17 | Sampled audio file 2 |
| ringer17 | Ring18 | Sampled audio file 3 |
| ringer18 | Ring19 | Sampled audio file 4 |
| ringer19 | Ring20 | Sampled audio file 5 |
| ringer20 | Ring21 | Sampled audio file 6 |
| ringer21 | Ring22 | Sampled audio file 7 |
| ringer22 | Ring23 | Sampled audio file 8 |
| ringer23 | Ring24 | Sampled audio file 9 |
| ringer24 | Ring25 | Sampled audio file 10 |



Note: Silent Ring

Silent ring will provide a visual indication of an incoming call, but no audio indication. Sampled audio files 1 to 10 all use the same built-in file unless that file has been replaced with a downloaded file. For more information, see [<saf/>](#).

Miscellaneous Patterns

The following table shows the miscellaneous patterns and their descriptions:

Table 13-75: Miscellaneous Pattern Names

| <i>Miscellaneous pattern name</i> | <i>Description</i> |
|-----------------------------------|--------------------------------|
| instant message | New instant message |
| local hold notification | Local hold notification |
| message waiting | New message waiting indication |
| negative confirmation | Negative confirmation |
| positive confirmation | Positive confirmation |
| remote hold notification | Remote hold notification |
| welcome | Welcome (boot up) |

<rt/>

Ringtone is used to define a simple class of ring to be applied based on some credentials that are usually carried within the network protocol. The ring class includes parameters such as call-waiting and ringer index, if appropriate. The ring class can use one of four types of ring that are defined as follows:

ring Play a specified ring pattern or call waiting indication

visual Provide only a visual indication (no audio) of an incoming call, no ringer needs to be specified

answer Provide auto-answer on an incoming call

ring-answer Provide auto-answer on an incoming call after a certain number of rings

**Note: Using the Answer Ring Type**

The auto-answer on incoming call is currently only applied if there is no other call in progress on the phone at the time.

The phone supports the following ring classes: **default**, **visual**, **answerMute**, **autoAnswer**, **ringAnswerMute**, **ringAutoAnswer**, **internal**, **external**, **emergency**, **precedence**, **splash**, and **custom<y>** where y is 1 to 17.

In the following table, x is the ring class name.


Caution: Ringtone Parameters Will Not Work After a Software Downgrade

If a phone has been upgraded to Polycom UC Software 4.0.0 and then downgraded to SIP 3.2.3 or earlier, the ringtone parameters will be unusable due to configuration parameters name changes in UC Software 4.0.0.

Table 13-76: Sound Effects Ringtone Parameters

| <i>Parameter</i> | <i>Permitted Values</i> |
|---|---|
| se.rt.enabled | 0 or 1 (default) |
| If 0 , the ringtone feature is not enabled on the phone. If 1 (default), the ringtone feature is enabled. | |
| se.rt.modification.enabled | 0 or 1 (default) |
| A flag to determine whether or not to allow user modification (through phone's user interface) of the pre-defined ringtone enabled for modification. | |
| se.rt.<ringClass>.callWait | callWaiting, callWaitingLong, precedenceCallWaiting |
| The call waiting tone to be used for this class of ring. The call waiting should match one defined in Table 14-591: Call Progress Tone Pattern Names. The default call waiting tone is <code>callWaiting</code> . | |
| se.rt.<ringClass>.name | UTF-8 encoded string |
| The answer mode for a ringtone. Used for identification purposes in the user interface. | |
| se.rt.<ringClass>.ringer | default, ringer1 to ringer24 |
| The ringtone to be used for this class of ring. The ringer should match one of Table 14-60: Ringtone Pattern Names . The default ringer is <code>ringer2</code> . | |
| se.rt.<ringClass>.timeout | 1 to 60000 only relevant if the type is set to ring-answer |
| The duration of the ring in milliseconds before the call is auto answered. The default is 2000. | |
| se.rt.<ringClass>.type | ring, visual, answer, ring-answer |
| The answer mode for a ringtone as defined in list earlier in this section. | |

<sec/>

This parameter affects the security features of the phone. The configuration parameter is defined as follows:

Table 13-77: General Security Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|------------------------------------|-------------------------|----------------|
| sec.tagSerialNo¹ | 0 or 1 | 0 |

If 0, the phone does not advertise its serial number (MAC address) through protocol signaling. If 1, the phone may advertise its serial number through protocol signaling.

¹ Change causes phone to restart or reboot.

This parameter also includes:

- <encryption/>
- <pwd/><length/>
- <srtplib/>
- <H235/>
- <dot1x><eapollogoff/>
- <hostmovedetect/>
- <TLS/>

<encryption/>

This configuration parameter is defined as follows:

Table 13-78: File Encryption Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| sec.encryption.upload.callLists¹ | 0 or 1 | 0 |

The encryption on the phone-specific call lists that is uploaded to the provisioning server. If 0, the call list is uploaded unencrypted regardless of how it was downloaded, the directory replaces whatever phone-specific call list is on the server, even if the file on the server is encrypted. If 1, the call list is uploaded encrypted regardless of how it was downloaded. The file replaces any existing phone-specific call lists file on the server.

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| sec.encryption.upload.config | 0 or 1 | 0 |
| <p>The encryption on the phone-specific configuration file created and uploaded to the provisioning server when the user selects Upload Configuration from the phone menu.</p> <p>If 0, the file is uploaded unencrypted, and overwrites whatever phone-specific configuration file is on the server, even if the file on the server is encrypted.</p> <p>If 1, the file is uploaded encrypted and replaces any existing phone-specific configuration file on the server. If there is no encryption key on the phone, the file is not uploaded.</p> | | |
| sec.encryption.upload.dir¹ | 0 or 1 | 0 |
| <p>The encryption on the phone-specific contact directory that is uploaded to the provisioning server.</p> <p>If 0, the directory is uploaded unencrypted regardless of how it was downloaded, the directory replaces whatever phone-specific contact directory is on the server, even if the file on the server is encrypted.</p> <p>If 1, the directory is uploaded encrypted regardless of how it was downloaded. The file replaces any existing phone-specific contact directory file on the server.</p> | | |
| sec.encryption.upload.overrides | 0 or 1 | 0 |
| <p>The encryption on the phone-specific <MACaddress>-phone.cfg override file that is uploaded to the server.</p> <p>If 0, the file is uploaded unencrypted regardless of how it was downloaded, the file replaces whatever file was on the server, even if the file on the server is encrypted.</p> <p>If 1, the file is uploaded encrypted regardless of how it was downloaded. The file replaces any existing phone-specific override file on the server.</p> | | |

¹ Change causes phone to restart or reboot.

<pwd/><length/>

This configuration parameter is defined as follows:

Table 13-79: Password Length Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| sec.pwd.length.admin¹ | 0-32 | 1 |
| <p>The minimum length for administrator passwords changed using the phone. Use 0 to allow null passwords.</p> | | |
| sec.pwd.length.user¹ | 0-32 | 2 |
| <p>The minimum length for user passwords changed using the phone. Use 0 to allow null passwords.</p> | | |

¹ Change causes phone to restart or reboot.

<srtp/>

As per RFC 3711, you cannot turn off authentication of RTCP. The configuration parameter is defined as follows:

Table 13-80: SRTP Parameters

| <i>Parameter</i> | <i>Permitted values</i> | <i>Defaults</i> |
|---|--|-----------------|
| sec.srtp.answerWithNewKey¹ | 0 or 1 | 1 |
| If 0, a new key is not provided when answering a call. If 1, a new key is provided when answering a call. | | |
| sec.srtp.enable² | 0 or 1 | 1 |
| If 0, the phone always declines SRTP offers. If 1, the phone accepts SRTP offers. <i>Note:</i> The defaults for SIP 3.2.0 was 0 when Null or not defined. | | |
| sec.srtp.holdWithNewKey¹ | 0 or 1 | 1 |
| If 0, a new key is not provided when holding a call. If 1, a new key is provided when holding a call. | | |
| sec.srtp.key.lifetime² | 0, positive integer
minimum 1024 or
power of 2 notation | Null |
| The lifetime of the master key used for the cryptographic parameter in SDP. The value specified is the number of SRTP packets. If 0, the master key lifetime is not set. If set to a valid value (at least 1024, or a power such as 2^{10}), the master key lifetime is set. When the lifetime is set, a re-invite with a new key will be sent when the number or SRTP packets sent for an outgoing call exceeds half the value of the master key lifetime. <i>Note:</i> Setting this parameter to a non-zero value may affect the performance of the phone. | | |
| sec.srtp.mki.enabled² | 0 or 1 | 0 |
| The master key identifier (MKI) is an optional parameter for the cryptographic parameter in the SDP that uniquely identifies the SRTP stream within an SRTP session. MKI is expressed as a pair of decimal numbers in the form: mki:mki_length where mki is the MKI value and mki_length its length in bytes. If 1, a four-byte MKI parameter is sent within the SDP message of the SIP INVITE / 200 OK. If 0, the MKI parameter is not sent. | | |
| sec.srtp.mki.length² | 1 to 4 | 4 |
| The length of the master key identifier (MKI), in bytes. Microsoft Lync offers 1-byte MKIs. | | |
| sec.srtp.mki.startSessionAtOne | 0 or 1 | 0 |
| If set to 1, use an MKI value of 1 at the start of an SDP session. If set to 0, the MKI value will increment for each new crypto key. | | |
| sec.srtp.offer² | 0 or 1 | 0 |
| If 1, the phone includes a secure media stream description along with the usual non-secure media description in the SDP of a SIP INVITE. This parameters applies to the phone initiating (offering) a phone call. If 0, no secure media stream is included in SDP of a SIP invite. | | |

| <i>Parameter</i> | <i>Permitted values</i> | <i>Defaults</i> |
|--|-------------------------|-----------------|
| sec.srtp.offer.HMAC_SHA1_32² | 0 or 1 | 0 |
| If 1, a crypto line with the AES_CM_128_HMAC_SHA1_32 crypto-suite will be included in offered SDP. If 0, the crypto line is not included. | | |
| sec.srtp.offer.HMAC_SHA1_80² | 0 or 1 | 1 |
| If 1, a crypto line with the AES_CM_128_HMAC_SHA1_80 crypto-suite will be included in offered SDP. If 0, the crypto line is not included. | | |
| sec.srtp.padRtpToFourByteAlignment² | 0 or 1 | 0 |
| Packet padding may be required when sending or receiving video from other video products. If 1, RTP packet padding is needed. If 0, no packet padding is needed. | | |
| sec.srtp.require² | 0 or 1 | 0 |
| If 0, secure media streams are not required. If 1, the phone is only allowed to use secure media streams. Any offered SIP INVITEs must include a secure media description in the SDP or the call will be rejected. For outgoing calls, only a secure media stream description is included in the SDP of the SIP INVITE, meaning that the non-secure media description is not included. If this parameter set to 1, <code>sec.srtp.offer</code> will also be set to 1, regardless of the value in the configuration file. | | |
| sec.srtp.requireMatchingTag² | 0 or 1 | 1 |
| If 0, the tag values in the crypto parameter in an SDP answer are ignored. If 1, the tag values must match. | | |
| sec.srtp.resumeWithNewKey¹ | 0 or 1 | 1 |
| If 0, a key is not provided when resuming a call. If 1, a key is provided when resuming a call. | | |
| sec.srtp.sessionParams.noAuth.offer² | 0 or 1 | 0 |
| If 0, authentication of RTP is offered. If 1, no authentication of RTP is offered; a session description that includes the UNAUTHENTICATED_SRTP session parameter is sent when initiating a call. | | |
| sec.srtp.sessionParams.noAuth.require² | 0 or 1 | 0 |
| If 0, authentication of RTP is required. If 1, no authentication of RTP is required; a call placed to a phone configured with this parameter must offer the UNAUTHENTICATED_SRTP session parameter in its SDP. If this parameter is set to 1, <code>sec.srtp.sessionParams.noAuth.offer</code> will also be set to 1, regardless of the value in the configuration file. | | |
| sec.srtp.sessionParams.noEncryptRTCP.offer² | 0 or 1 | 0 |
| If 0, encryption of RTCP is offered. If 1, no encryption of RTCP is offered; a session description that includes the UNENCRYPTED_SRTCP session parameter is sent when initiating a call. | | |
| sec.srtp.sessionParams.noEncryptRTCP.require² | 0 or 1 | 0 |
| If set to 0, encryption of RTCP is required. If set to 1, no encryption of RTCP is required; a call placed to a phone configured with <code>noAuth.require</code> must offer the UNENCRYPTED_SRTCP session parameter in its SDP. If this parameter is set to 1, <code>sec.srtp.sessionParams.noEncryptRTCP.offer</code> will also be set to 1, regardless of the value in the configuration file. | | |

| <i>Parameter</i> | <i>Permitted values</i> | <i>Defaults</i> |
|---|-------------------------|-----------------|
| sec.srtp.sessionParams.noEncryptRTP.offer² | 0 or 1 | 0 |
| If 0, encryption of RTP is offered. If 1, no encryption of RTP is offered; a session description that includes the UNENCRYPTED_SRTP session parameter is sent when initiating a call. | | |
| sec.srtp.sessionParams.noEncryptRTP.require² | 0 or 1 | 0 |
| If 0, encryption of RTP is required. If 1, no encryption of RTP is required. A call placed to a phone configured with noAuth.require must offer the UNENCRYPTED_SRTP session parameter in its SDP. If set to 1, sec.srtp.sessionParams.noEncryptRTP.offer will also be set to 1, regardless of the value in the configuration file. | | |
| sec.srtp.simplifiedBestEffort | 0 or 1 | 0 |
| If 0, no SRTP is supported. If 1, negotiation of SRTP compliant with Microsoft Session Description Protocol Version 2.0 Extensions is supported. | | |

¹ Supported on only the SpectraLink handsets.

² Change causes phone to restart or reboot.

<H235/>

At this time, this parameter is used with the Polycom VVX 1500 phone only. The H.235 Voice Profile implementation is Polycom HDX compatible. OpenSSL-based Diffie-Hellman key exchange and AES-128 CBC encryption algorithms are used to encrypt the RTP media.

The configuration parameter is defined as follows:

Table 13-81: H.235 Media Encryption Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| sec.H235.mediaEncryption.enabled¹ | 0 or 1 | 1 |
| If 0, H.235 Voice Profile RTP media encryption will be disabled. If 1, H.235 media encryption will be enabled and negotiated when such encryption is requested by the far end. | | |
| sec.H235.mediaEncryption.offer¹ | 0 or 1 | 0 |
| If 0, media encryption negotiations will not be initiated with the far end. If 1 and sec.H235.mediaEncryption.enabled is also 1, media encryption negotiations will be initiated with the far end; however, successful negotiations are not a requirement for the call to complete. | | |
| sec.H235.mediaEncryption.require¹ | 0 or 1 | 0 |
| If 0, media encryption negotiations will not be required. If 1 and sec.H235.mediaEncryption.enabled is also 1, media encryption negotiations will be initiated or completed with the far end, and if negotiations fail, the call will be dropped. | | |

¹ Change causes phone to restart or reboot.

<dot1x><eapollogoff/>

This configuration parameter is defined as follows:

Table 13-82: 802.1X EAP over LAN (EAPOL) Logoff Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| sec.dot1x.eapollogoff.enabled¹ | 0 or 1 | 0 |
| If 0, the phone will not send an EAPOL Logoff message on behalf of the disconnected supplicant. If 1, the feature is enabled and the phone will send an EAPOL Logoff message on behalf of the disconnected supplicant connected to the phone's secondary (PC) port. | | |
| sec.dot1x.eapollogoff.lanlinkreset¹ | 0 or 1 | 0 |
| If 0, the phone software will not reset (recycle) the LAN port link in the application initiation stage. If 1, the LAN port link will be reset in the application initiation stage. | | |

¹ Change causes phone to restart or reboot.

<hostmovedetect/>

This configuration parameter is defined as follows:

Table 13-83: Host Movement Detection Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| sec.hostmovedetect.cdp.enabled¹ | 0 or 1 | 0 |
| If set to 1, the phone software will unconditionally send a CDP packet (to the authenticator switch port) to indicate a host has been connected or disconnected to its secondary (PC) port. | | |
| sec.hostmovedetect.cdp.sleepTime¹ | 0 to 60000 | 1000 |
| If <code>sec.hostmovedetect.cdp.enabled</code> is set to 1, then there will be an x microsecond time interval between two consecutive link-up state change reports. This will reduce the frequency of dispatching CDP packets. | | |

¹ Change causes phone to restart or reboot.

<TLS/>

For the list of configurable ciphers, see [Configurable TLS Cipher Suites](#).

This parameter also includes [<profile/>](#) and [<profileSelection/>](#).

Table 13-84: TLS Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|---|
| sec.TLS.browser.cipherList
The cipher list for browser. | String | NoCipher |
| sec.TLS.cipherList
The global cipher list parameter. | String | “RSA:!EXP:!LOW:!NULL:!MD5:@STRENGTH” |
| sec.TLS.customCaCert.x
The custom certificate for TLS Application Profile x (x= 1 to 6). | String | Null |
| sec.TLS.customDeviceCert.x
The custom device certificate for TLS Application Profile x (x= 1 to 6). | String | Null |
| sec.TLS.customDeviceKey.x
The custom device certificate private key for TLS Application Profile x (x= 1 to 6). | String | Null |
| sec.TLS.LDAP.cipherList
The cipher list for the corporate directory. | String | NoCipher |
| sec.TLS.prov.cipherList
The cipher list for provisioning. | String | NoCipher |
| sec.TLS.SIP.cipherList
The cipher list for SIP. | String | NoCipher |
| sec.TLS.SIP.strictCertCommonNameValidation
If 1, enable common name validation for SIP. | 0 or 1 | 1 |
| sec.TLS.syslog.cipherList
The cipher list for syslog. | String | NoCipher |
| sec.TLS.xmpp.cipherList
The cipher list for CMA presence. | String | NoCipher |

<profile/>

Profiles are a collection of related security parameters. There are two platform profiles and six application profiles.

Table 13-85: TLS Profile Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|----------------|
| sec.TLS.profile.x.caCert.application1
Application CA 1 | 0 or 1 | 1 |
| sec.TLS.profile.x.caCert.application2
Application CA 2 | 0 or 1 | 1 |
| sec.TLS.profile.x.caCert.application3
Application CA 3 | 0 or 1 | 1 |
| sec.TLS.profile.x.caCert.application4
Application CA 4 | 0 or 1 | 1 |
| sec.TLS.profile.x.caCert.application5
Application CA 5 | 0 or 1 | 1 |
| sec.TLS.profile.x.caCert.application6
Application CA 6 | 0 or 1 | 1 |
| sec.TLS.profile.x.caCert.platform1
Platform CA 1 | 0 or 1 | 1 |
| sec.TLS.profile.x.caCert.platform2
Platform CA 2 | 0 or 1 | 1 |
| Specify which CA certificates should be used for TLS Application Profile x (where x is 1 to 6). If set to 0, the CA will not be used. If set to 1, the CA will be used. | | |
| sec.TLS.profile.x.caCert.defaultList | String | Null |
| The list of default CA certificates for TLS Application Profile x (x= 1 to 6). | | |
| sec.TLS.profile.x.cipherSuite | String | Null |
| The cipher suite for TLS Application Profile x (where x is 1 to 6). | | |
| sec.TLS.profile.x.cipherSuiteDefault | 0 or 1 | 1 |
| If 0, use the custom cipher suite for TLS Application Profile x (x= 1 to 6). If 1, use the default cipher suite. | | |
| sec.TLS.profile.x.deviceCert | Polycom, Platform1, Platform2, Application1, Application2, Application3, Application4, Application5, Application6 | Polycom |
| The device certificate to use for TLS Application Profile x (x = 1 to 6). | | |

<profileSelection/>

You can configure the following parameters to choose the platform profile or application profile to use for each TLS application.

The permitted values are:

- PlatformProfile1
- PlatformProfile2
- ApplicationProfile1
- ApplicationProfile2
- ApplicationProfile3
- ApplicationProfile4
- ApplicationProfile5
- ApplicationProfile6

Table 13-86: TLS Profile Selection Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---|-------------------------|
| sec.TLS.profileSelection.browser | a TLS profile | PlatformProfile1 |
| The TLS platform profile or TLS application profile (see preceding list) to use for the browser or microbrowser. | | |
| sec.TLS.profileSelection.LDAP | a TLS profile | PlatformProfile1 |
| The TLS platform profile or TLS application profile (see preceding list) to use for the Corporate Directory. | | |
| sec.TLS.profileSelection.SIP | a TLS profile | PlatformProfile1 |
| The TLS platform profile or TLS application profile (see preceding list) to use for SIP operations. | | |
| sec.TLS.profileSelection.syslog | PlatformProfile1 or PlatformProfile2 | PlatformProfile1 |
| The TLS platform profile to use for syslog operations. | | |
| sec.TLS.profileSelection.XMPP | a TLS profile | PlatformProfile1 |
| The TLS platform profile or TLS application profile (see preceding list) to use for the CMA Directory. | | |

<softkey/>

Note that `feature.enhancedFeatureKeys.enabled` must be enabled (set to 1) to use the Configurable Soft Key feature.

The configuration parameter is defined as follows (where x=1 to a maximum number of defined soft keys).

Table 13-87: Soft Key Customization Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| softkey.feature.basicCallManagement.redundant | 0 or 1 | 1 |
| Control the display of the Hold , Transfer , and Conference soft keys. If set to 0 and the phone has hard keys mapped for Hold , Transfer , and Conference functions (all must be mapped), none of the soft keys are displayed. If set to 1, all of these soft keys are displayed. | | |
| softkey.feature.buddies | 0 or 1 | 1 |
| If 0, the Buddies soft key is not displayed. If 1, the soft key is displayed (if <code>pres.idleSoftKeys</code> is set to 1). | | |
| softkey.feature.callers | 0 or 1 | 0 |
| If 1, the Callers soft key displays on all platforms. If 0, the Callers soft key is disabled for all platforms. The default value for the SoundPoint IP 321/331/335 is 1. For all other platforms, the default is 0.
Note: Using a NULL value will result in different behaviour on SoundPoint IP 321/331/335 than on other platforms because of the different default value. | | |
| softkey.feature.directories | 0 or 1 | 0 |
| If 1, the Dir soft key displays on all platforms. If 0, the Dir soft key is disabled for all platforms. The default value for the SoundPoint IP 321/331/335 is 1. For all other platforms, the default is 0.
Note: Using a NULL value will result in different behaviour on SoundPoint IP 321/331/335 than on other platforms because of the different default value. | | |
| softkey.feature.endcall | 0 or 1 | 1 |
| If 0, the End Call soft key is not displayed. If 1, the soft key is displayed. | | |
| softkey.feature.forward | 0 or 1 | 1 |
| If 0, the Forward soft key is not displayed. If 1, the soft key is displayed. <i>Note:</i> For the SoundPoint IP 321/331/335 phones, you must create the soft key using the Enhanced Feature Key feature to display it. | | |
| softkey.feature.join | 0 or 1 | 1 |
| Join two individual calls to form a conference. If 0, the Join soft key is not displayed. If 1, the soft key is displayed. | | |
| softkey.feature.mystatus | 0 or 1 | 1 |
| If 0, the MyStatus soft key is not displayed. If 1, the soft key is displayed (if <code>pres.idleSoftKeys</code> is set to 1). | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|----------------|
| softkey.feature.newcall | 0 or 1 | 1 |
| If 0, the New Call soft key is not displayed when there is an alternative way to place a call. If 1, the New Call soft key is displayed. | | |
| softkey.feature.simplifiedSignIn | 0 or 1 | 0 |
| If 0, the SignIn soft key is not displayed. If 1 and <code>voIpProt.server.x.specialInterop</code> is <code>lync2010</code> , the SignIn soft key is displayed. | | |
| softkey.feature.split | 0 or 1 | 1 |
| Split up a conference into individual calls. If 0, the Split soft key is not displayed. If 1, the soft key is displayed. | | |
| softkey.x.action | macro action string, 256 characters | Null |
| The action or function for custom soft key x. This value uses the same macro action string syntax as an Enhanced Feature Key. For a list of actions, see Understanding Macro Definitions. | | |
| softkey.x.enable | 0 or 1 | 0 |
| If 0, the soft key x is disabled. If 1, the soft key is enabled. | | |
| softkey.x.insert | 0 to 10 | 0 |
| The position on the phone screen for soft key x. For example, if the value is 3, the soft key will be displayed on the screen in the third position from the left. <i>Note:</i> If <code>softkey.x.precede</code> is configured, this value is ignored. If the insert location is greater than the number of soft keys, the key will be positioned last, after the other soft keys. | | |
| softkey.x.label | string | Null |
| The text displayed on the soft key label. If Null, the label is determined as follows: <ul style="list-style-type: none"> • If the soft key performs an Enhanced Feature Key macro action, the label of the macro will be used. • If the soft key calls a speed dial, the label of the speed dial contact will be 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 will be blank. | | |
| softkey.x.precede | 0 or 1 | 0 |
| If 0, soft key x is positioned in the first empty space from the left. If 1, the soft key is displayed before (to the left of) the first default soft key. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| softkey.x.use.active
Display in the active call state | 0 or 1 | 0 |
| softkey.x.use.alerting
Display in the alerting state | 0 or 1 | 0 |
| softkey.x.use.dialtone
Display in the dial tone state | 0 or 1 | 0 |
| softkey.x.use.hold
Display in the hold state | 0 or 1 | 0 |
| softkey.x.use.idle
Display in the idle state | 0 or 1 | 0 |
| softkey.x.use.proceeding
Display in the proceeding state | 0 or 1 | 0 |
| softkey.x.use.setup
Display in the proceeding state | 0 or 1 | 0 |

If 0, the soft key is not displayed when the phone is in the call state. If 1, the soft key is displayed when the phone is in the call state.

<tcplpApp/>

This parameter includes:

- <dhcp/>
- <dns/>
- <ice/>
- <sntp/>
- <port/><rtp/>
- <keepalive/>
- <fileTransfer/>

<dhcp/>

The DHCP parameters enable you to change how the phone reacts to DHCP changes.

Table 13-88: DHCP Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| tcplpApp.dhcp.releaseOnLinkRecovery | 0 or 1 | 1 |
| If 0, no DHCP release occurs. If 1, a DHCP release is performed after the loss and recovery of the network. | | |

<dns/>

The <dns/> parameter provides a way to set Domain Name System (DNS). However, any values set through DHCP will have a higher priority and any values set through the <device/> parameter in a configuration file will have a lower priority.

Table 13-89: Domain Name System (DNS) Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|----------------------------------|----------------|
| tcplpApp.dns.server¹ | Dotted-decimal IP address | Null |
| The primary server to which the phone directs DNS queries. | | |
| tcplpApp.dns.altServer¹ | Dotted-decimal IP address | Null |
| The secondary server to which the phone directs DNS queries. | | |
| tcplpApp.dns.domain¹ | String | Null |
| The phone's DNS domain. | | |

¹ Change causes phone to restart or reboot.

<ice/>

The <ice/> parameter enables you to set the STUN/TURN/ICE feature.

Table 13-90: Ice Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|----------------------------------|-----------------|
| tcplpApp.ice.mode | Disabled, Standard, MSOCS | Disabled |
| Turn SIP ICE negotiation on or off. If using Lync Server 2010, set to MSOCS to enable ICE. | | |
| tcplpApp.ice.password | String | Null |
| Enter the password to authenticate to the TURN server. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| tcplpApp.ice.stun.server | String | Null |
| Enter the IP address of the STUN server. | | |
| tcplpApp.ice.stun.udpPort | 1-65535 | 3478 |
| The UDP port number of the STUN server. | | |
| tcplpApp.ice.tcp.enabled | 0 or 1 | 1 |
| If 0, TCP is disabled. If 1, TCP is enabled. | | |
| tcplpApp.ice.turn.callAdmissionControl.enabled | | 1 |
| tcplpApp.ice.turn.server | String | Null |
| Enter the IP address of the TURN server. | | |
| tcplpApp.ice.turn.tcpPort | 1-65535 | 443 |
| The UDP port number of the TURN server. | | |
| tcplpApp.ice.turn.udpPort | 1-65535 | 443 |
| The UDP port number of the TURN server. | | |
| tcplpApp.ice.username | String | Null |
| Enter the user name to authenticate to the TURN server. | | |

<snntp/>

The following table describes the Simple Network Time Protocol (SNTP) parameters used to set up time synchronization and daylight savings time. The default values will enable and configure daylight savings time (DST) for North America.

Daylight savings time defaults:

- Do not use fixed day, use first or last day of week in the month.
- Start DST on the second Sunday in March at 2am.
- Stop DST on the first Sunday in November at 2am.

Table 13-91: Simple Network Time Protocol (SNTP) Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------------------|------------------|
| tcplpApp.snntp.address | Valid hostname or IP address | Null |
| The address of the SNTP server. | | |
| tcplpApp.snntp.address.overrideDHCP | 0 or 1 | 0 |
| If 0, the DHCP values for the SNTP server address will be used. If 1, the SNTP parameters will override the DHCP values. | | |
| tcplpApp.snntp.daylightSavings.enable | 0 or 1 | 1 |
| If 0, daylight savings time rules are not applied to the displayed time. If 1, the daylight savings rules apply. | | |
| tcplpApp.snntp.daylightSavings.fixedDayEnable | 0 or 1 | 0 |
| If 0, month, date, and dayOfWeek are used in the DST calculation. If 1, only month and date are used. | | |
| tcplpApp.snntp.daylightSavings.start.date | 1 to 31 | 8 |
| The start date for daylight savings time. If <code>fixedDayEnable</code> is set to 1, the value of this parameter is the day of the month to start DST. If <code>fixedDayEnable</code> is set to 0, this value specifies the occurrence of <code>dayOfWeek</code> when DST should start. Set 1 for the first occurrence in the month, set 8 for the second occurrence, 15 for the third occurrence, or 22 for the fourth occurrence. For example, if set to 15, DST starts on the third <code>dayOfWeek</code> of the month. | | |
| tcplpApp.snntp.daylightSavings.start.dayOfWeek | 1 to 7 | 1 |
| The day of the week to start DST. 1=Sunday, 2=Monday, ... 7=Saturday. <i>Note:</i> this parameter is not used if <code>fixedDayEnable</code> is set to 1. | | |
| tcplpApp.snntp.daylightSavings.start.dayOfWeek.lastInMonth | 0 or 1 | 0 |
| If 1, DST starts on the last <code>dayOfWeek</code> of the month and the <code>start.date</code> is ignored). <i>Note:</i> this parameter is not used if <code>fixedDayEnable</code> is set to 1. | | |
| tcplpApp.snntp.daylightSavings.start.month | 1 to 12 | 3 (March) |
| The month to start DST. 1=January, 2=February... 12=December. | | |
| tcplpApp.snntp.daylightSavings.start.time | 0 to 23 | 2 |
| The time of day to start DST – in 24 hour clock format. 0= 12am, 1= 1am,... 12= 12pm, 13= 1pm, ... 23= 11pm. | | |
| tcplpApp.snntp.daylightSavings.stop.date | 1 to 31 | 1 |
| The stop date for daylight savings time. If <code>fixedDayEnable</code> is set to 1, the value of this parameter is the day of the month to stop DST. If <code>fixedDayEnable</code> is set to 0, this value specifies the occurrence of <code>dayOfWeek</code> when DST should stop. Set 1 for the first occurrence in the month, set 8 for the second occurrence, 15 for the third occurrence, or 22 for the fourth occurrence. For example, if set to 22, DST stops on the fourth <code>dayOfWeek</code> of the month. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------------------|----------------|
| tcplpApp.snntp.daylightSavings.stop.dayOfWeek | 1 to 7 | 1 |
| The day of the week to stop DST. 1=Sunday, 2=Monday, ... 7=Saturday. <i>Note:</i> this parameter is not used if <code>fixedDayEnable</code> is set to 1. | | |
| tcplpApp.snntp.daylightSavings.stop.dayOfWeek.lastInMonth | 0 or 1 | 0 |
| If 1, DST stops on the last <code>dayOfWeek</code> of the month and the <code>stop.date</code> is ignored). <i>Note:</i> this parameter is not used if <code>fixedDayEnable</code> is set to 1. | | |
| tcplpApp.snntp.daylightSavings.stop.month | 1 to 12 | 11 |
| The month to stop DST. 1=January, 2=February... 12=December. | | |
| tcplpApp.snntp.daylightSavings.stop.time | 0 to 23 | 2 |
| The time of day to stop DST – in 24 hour clock format. 0= 12am, 1= 1am,... 12= 12pm, 13= 1pm, ... 23= 11pm. | | |
| tcplpApp.snntp.gmtOffset | positive or negative integer | 0 |
| The offset in seconds of the local time zone from GMT.3600 seconds = 1 hour, -3600 seconds = -1 hour. | | |
| tcplpApp.snntp.gmtOffset.overrideDHCP | 0 or 1 | 0 |
| If 0, the DHCP values for the GMT offset will be used. If 1, the SNTP values for the GMT offset will be used. | | |
| tcplpApp.snntp.resyncPeriod | positive integer | 86400 |
| The period of time (in seconds) that passes before the phone resynchronizes with the SNTP server. <i>Note:</i> 86400 seconds is 24 hours. | | |

<port/><rtp/>

These parameters allow you to configure the port filtering used for RTP traffic.

Table 13-92: RTP Port Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| tcplpApp.port.rtp.filterByIp¹ | 0 or 1 | 1 |
| IP addresses can be negotiated through the SDP or H.323 protocols. If set to 1, the phone rejects RTP packets that arrive from non-negotiated IP addresses. <i>Note:</i> the H.323 protocol is supported on only the VVX 1500 phones. | | |
| tcplpApp.port.rtp.filterByPort¹ | 0 or 1 | 0 |
| Ports can be negotiated through the SDP protocol. If set to 1, the phone will reject RTP packets arriving from (sent from) a non-negotiated port. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-----------------------------------|----------------|
| tcpIpApp.port.rtp.forceSend¹ | 0 to 65535 | 0 |
| <p>Send all RTP packets to, and expect all RTP packets to arrive on, this port. If 0, RTP traffic is not forced to one port. <i>Note:</i> Both <code>tcpIpApp.port.rtp.filterByIp</code> and <code>tcpIpApp.port.rtp.filterByPort</code> must be set to 1 for this to work.</p> | | |
| tcpIpApp.port.rtp.mediaPortRangeStart¹ | even integer 1024 to 65440 | 2222 |
| <p>The starting port for RTP packets. Ports will be allocated from a pool starting with this port up to a value of (start-port + 47) for a voice-only phone or (start-port + 95) for a video phone.
<i>Note:</i> Ensure that there is no contention for port numbers. For example, do not use 5060 (default port for SIP).</p> | | |

¹ Change causes phone to restart or reboot.

<keepalive/>

This parameter enables the configuration of TCP keep-alive on SIP TLS connections; the phone can detect a failure quickly (in minutes) and attempt to re-register with the SIP call server (or its redundant pair).

Table 13-93: TCP Keep-Alive Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| tcpIpApp.keepalive.tcp.idleTransmitInterval | 10 to 7200 | 30 |
| <p>The amount of time to wait (in seconds) before sending the keep-alive message to the call server.
<i>Note:</i> If this parameter is set to a value that is out of range, the default value is used.
<i>Note:</i> On the VVX 1500 phone, <code>tcpIpApp.keepalive.tcp.idleTransmitInterval</code> is the number of seconds TCP waits between transmission of the last data packet and the first keep-alive message.</p> | | |
| tcpIpApp.keepalive.tcp.noResponseTransmitInterval | 5 to 120 | 20 |
| <p>If no response is received to a keep-alive message, subsequent keep-alive messages are sent to the call server at this interval (every x seconds).
<i>Note:</i> On the VVX 1500 phone, this parameter specifies the amount of idle time between the transmission of the keep-alive packets the TCP stack waits. This applies whether the last keep-alive was acknowledged or not.</p> | | |
| tcpIpApp.keepalive.tcp.sip.tls.enable | 0 or 1 | 0 |
| <p>If 0, disable TCP keep-alive for SIP signaling connections that use TLS transport. If 1, enable TCP keep-alive for SIP signaling connections that use TLS transport.</p> | | |

¹ Change causes phone to restart or reboot.

<fileTransfer/>

The <fileTransfer/> parameter provides information on file transfers from the phone to the Provisioning server.

Table 13-94: File Transfer Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| tcplpApp.fileTransfer.waitForLinkIfDown | 0 or 1 | 1 |
| <p>If 1, file transfer from the FTP server is delayed until Ethernet comes back up.
 If 0, file transfer from the FTP server is not attempted.
 This flag is set to 0 when the SoundStation Duo is in PSTN mode. File transfer does not happen; the file embedded in the software package is used.</p> | | |

<tones/>

This parameter describes configuration items for the tone resources available in the phone. It includes:

- <DTMF/>
- <chord/>

<DTMF/>

This configuration parameter is defined as follows:

Table 13-95: DTMF Tone Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| tone.dtmf.chassis.masking¹ | 0 or 1 | 0 |
| <p>If 0, DTMF tones will be played through the speakerphone in handsfree mode. If 1 (set only if <code>tone.dtmf.viaRtp</code> is set to 0), DTMF tones will be substituted with non-DTMF pacifier tones when dialing in handsfree mode – this is to prevent the tones from broadcasting to surrounding telephony devices or being inadvertently transmitted in-band due to local acoustic echo.</p> | | |
| tone.dtmf.level¹ | -33 to 3 | -15 |
| <p>The level of the high frequency component of the DTMF digit measured in dBm0; the low frequency tone will be two dB lower.</p> | | |
| tone.dtmf.offTime¹ | positive integer | 50 |
| <p>When a sequence of DTMF tones is played out automatically, this is the length of time in milliseconds the phone will pause between digits. This is also the minimum inter-digit time when dialing manually.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| tone.dtmf.onTime¹ | positive integer | 50 |
| When a sequence of DTMF tones is played out automatically, this is the length of time in milliseconds the tones will be played for. This is also the minimum time the tone will be played when dialing manually (even if key press is shorter). | | |
| tone.dtmf.rfc2833Control¹ | 0 or 1 | 1 |
| If set to 1, the phone will indicate a preference for encoding DTMF through RFC 2833 format in its Session Description Protocol (SDP) offers by showing support for the phone-event payload type. This does not affect SDP answers; these will always honor the DTMF format present in the offer since the phone has native support for RFC 2833. | | |
| tone.dtmf.rfc2833Payload¹ | 96 to 127 | 127 |
| The phone-event payload encoding in the dynamic range to be used in SDP offers. | | |
| tone.dtmf.viaRtp¹ | 0 or 1 | 1 |
| If set to 1, encode DTMF in the active RTP stream. Otherwise, DTMF may be encoded within the signaling protocol only when the protocol offers the option. <i>Note:</i> If this parameter is set to 0, <code>tone.dtmf.chassis.masking</code> should be set to 1. | | |

¹ Change causes phone to restart or reboot.

<chord/>

Chord-sets are the building blocks of sound effects that used synthesized audio rather than sampled audio. Most call progress and ringer sound effects are synthesized. A chord-set is a multi-frequency note with an optional on/off cadence. A chord-set can contain up to four frequency components generated simultaneously, each with its own level.

There are three chord sets: `callProg`, `misc`, and `ringer`. Each chord set has different chord names, represented by `x` in the following table. The chord names are as follows:

For **callProg**, `x` can be one of the following chords:

- **dialTone, busyTone, ringback, reorder, stutter_3, callWaiting, callWaitingLong, howler, recWarning, stutterLong, intercom, callWaitingLong, precedenceCallWaiting, preemption, precedenceRingback, or spare1 to spare6.**

For **misc**, `x` can be one of the following chords

- **spare1 to spare9.**

For **ringer**, `x` can be one of the following chords:

- **ringback, originalLow, originalHigh, or spare1 to spare19.**

Table 13-96: Chord Parameters

| <i>Parameter</i> | <i>Permitted Values</i> |
|---|-------------------------|
| tone.chord.callProg.x.freq.y | 0-1600 |
| tone.chord.misc.x.freq.y | 0-1600 |
| tone.chord.ringer.x.freq.y | 0-1600 |
| The frequency (in Hertz) for component y. Up to six chord-set components can be specified (y=1 to 6). | |
| tone.chord.callProg.x.level.y | -57 to 3 |
| tone.chord.misc.x.level.y | -57 to 3 |
| tone.chord.ringer.x.level.y | -57 to 3 |
| The level of component y in dBm0. Up to six chord-set components can be specified (y=1 to 6). | |
| tone.chord.callProg.x.onDur | positive integer |
| tone.chord.misc.x.onDur | positive integer |
| tone.chord.ringer.x.onDur | positive integer |
| The on duration (length of time to play each component) in milliseconds, 0=infinite. | |
| tone.chord.callProg.x.offDur | positive integer |
| tone.chord.misc.x.offDur | positive integer |
| tone.chord.ringer.x.offDur | positive integer |
| The off duration (the length of silence between each chord component) in milliseconds, 0=infinite. | |
| tone.chord.callProg.x.repeat | positive integer |
| tone.chord.misc.x.repeat | positive integer |
| tone.chord.ringer.x.repeat | positive integer |
| The number of times each ON/OFF cadence is repeated, 0=infinite. | |

<up/>

This per-site configuration is defined as follows:

Table 13-97: User Preferences Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| up.25mm | 1 or 2 | 1 |
| Specify whether to use a mobile phone or a PC to connect to the 2.5mm audio port on a conference phone. Set to 1 if using a mobile phone. Set to 2 if using a PC. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| up.accessibilityFeatures | 0 or 1 | 0 |
| VVX 1500 only. If 0, accessibility features are disabled. If 1, the screen background flashes orange for incoming calls. | | |
| up.analogHeadsetOption | 0, 1, or 3 | 0 |
| The Electronic Hookswitch mode for the phone's analog headset jack. 0 – no EHS-compatible headset is attached. 1 – a Jabra EHS-compatible headset is attached. 2 – a Plantronics EHS-compatible headset is attached. 3 – a Sennheiser EHS-compatible headset is attached. | | |
| up.audioMode | 0 or 1 | 0 |
| If 0, a handset is connected. If 1, a headset is connected. | | |
| up.audioSetup.auxInput | 0, 1, or 2 | 2 |
| SoundStation IP phones only. The auxiliary audio input. 0 – Other Input, 1 – Polycom Wireless Mic, 2 – off. | | |
| up.audioSetup.auxOutput | 0, 1, or 2 | 2 |
| SoundStation IP phones only. The auxiliary audio output. 0 – Other Input, 1 – Polycom Wireless Mic, 2 – off. | | |
| up.backlight.idleIntensity | 0, 1, 2, or 3 | 1 |
| The brightness of the LCD backlight when the phone is idle. 0 – off, 1 – low, 2 – medium, 3 – high. <i>Note:</i> If this is higher than the active backlight brightness (<i>onIntensity</i>), the active backlight brightness is used. | | |
| up.backlight.onIntensity | 0, 1, 2, or 3 | 3 |
| The brightness of the LCD backlight when the phone is active (in use). 0: off, 1 – low, 2 – medium, 3 – high | | |
| up.backlight.timeout | 5 to 60 | 40 |
| The number of seconds to wait before the backlight dims from the active intensity to the idle intensity. <i>Note:</i> the default for the SpectraLink handsets is 10 seconds. | | |
| up.cfgWarningsEnabled | 0 or 1 | 0 |
| If 1, a warning is displayed on the phone if the phone is configured with pre-UC software 3.3.0 parameters. If 0, the warning will not display. | | |
| up.handsfreeMode | 0 or 1 | 1 |
| If 0, the handsfree speakerphone is disabled (cannot be used). If 1, the handsfree speakerphone is enabled. | | |
| up.headsetAlwaysUseIntrinsicRinger | 0 or 1 | 1 |
| If 1, the USB headset will use the intrinsic ringer mixed with DSP ringer when the sound effect destination is the USB headset. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|------------------------------|----------------|
| up.headsetMode | 0 or 1 | 0 |
| If 0, handsfree mode will be used by default instead of the handset. If 1, the headset will be used as the preferred audio mode after the headset key is pressed for the first time, until the headset key is pressed again. | | |
| up.headsetOnlyAlerting | 0 or 1 | 0 |
| SpectralLink handsets only. If 1, only an auxiliary or Wi-Fi headset is used for alerting (such as incoming call alerting). | | |
| up.headset.phoneVolumeControl¹ | disable, enable, auto | auto |
| When a headset is connected to the phone, the phone's behavior with respect to volume control events from certain headsets is different.
enable – The phone responds to volume up/down events from the headset by displaying the volume widget in the phone's user interface and adjusting the phone's internal volume.
disable – The phone shall ignore volume up/down events from the headset; pressing the headset's volume controls has no effect on the phone.
auto – The phone shall automatically select which of the above two behaviors to apply, based upon the type and model version of headset that is attached. | | |
| up.hearingAidCompatibility.enabled | 0 or 1 | 0 |
| If set to 1, the phone audio Rx (receive) equalization is disabled for hearing aid compatibility. If 0, audio Rx equalization is enabled. | | |
| up.idleBrowser.enabled | 0 or 1 | 0 |
| If 0, the idle browser is disabled. If 1, the idle browser is enabled (if <code>up.prioritizeBackground.enable</code> is 1, the user can choose to display the background or the idle browser through the phone menu). | | |
| up.idleStateView¹ | 0 or 1 | 0 |
| Sets the default view on the phone.
If 0, The call/line view is the default view. If 1, the Home screen is the default view. | | |
| up.idleTimeout¹ | 0 to 65535, seconds | 40 |
| The number of seconds that the phone can be idle for before automatically leaving a menu and showing the idle display. If 0, there is no timeout and the phone does not automatically exit to the idle display. | | |
| up.lineKeyCallTerminate | 0 or 1 | 0 |
| If 1, the user can press a line key to end an active call on that line. If 0, the user cannot end a call by pressing the line key (this is the previous behavior). | | |
| up.localClockEnabled | 0 or 1 | 1 |
| If 0, the date and time are not shown on the idle display. If 1, the date and time and shown on the idle display. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---------------------------------|------------------|
| up.manualProtocolRouting | 0 or 1 | 1 |
| <p>VVX 1500 only. If 1, the user is presented with a protocol routing choice in situations where a call can be placed using either protocol (for example, with SIP and H.323 protocols). If 0, the default protocol is used and the user does not choose.</p> | | |
| up.manualProtocolRouting.softKeys | 0 or 1 | 1 |
| <p>Choose whether you want to display soft keys that control Manual Protocol Routing options. When Soft Key Control is enabled, you can use soft keys to choose between the SIP or H.323 protocol. When disabled, soft keys for protocol routing will not display. The soft keys are enabled by default.</p> | | |
| up.mwiVisible¹ | 0 or 1 | 0 |
| <p>If set is 0, the incoming MWI notifications for lines where the MWI callback mode is disabled (<code>msg.mwi.x.callBackMode</code> is set to 0) are ignored, and do not appear in the message retrieval menus. If set to 1, the MWI for lines whose MWI is disabled will display (pre-SIP 2.1 behavior), even though MWI notifications have been received for those lines.</p> | | |
| up.multiKeyAnswerEnabled | 0 or 1 | 0 |
| <p>SpectralLink 8400 series only. If 1, incoming calls can be answered by pressing any key. If 0, incoming calls can only be answered using the Talk button or the Start key.</p> | | |
| up.numberFirstCID¹ | 0 or 1 | 0 |
| <p>If 0, the caller ID display will show the caller's name first. If 1, the caller's phone number will be shown first.</p> | | |
| up.offHookAction.none¹ | 0 or 1 | 0 |
| <p>If 0, the behavior will be as it was in SIP 2.1.2. If 1, when the user lifts the handset, the phone will not seize the line and the ringer will continue until the user takes further action.</p> | | |
| up.oneTouchVoiceMail¹ | 0 or 1 | 0 |
| <p>If set to 1, the voicemail summary display is bypassed and voicemail is dialed directly (if configured).</p> | | |
| up.onHookDialingEnabled | 0 or 1 | 1 |
| <p>SpectralLink 8400 Series only. If 0, on hook dialing is disabled. If 1, on-hook dialing is enabled.</p> | | |
| up.operMode | 0 to 2, auto, PSTN, SPIP | 0 or auto |
| <p>Specifies the mode the SoundStation Duo phone will use.</p> <p>0—Auto (Automatic Mode Detect). The phone will automatically detect the mode to use, based on how the phone is set up. For more information, see PSTN Communications Settings.</p> <p>1—PSTN Only. The phone will operate in PSTN mode.</p> <p>2—SIP Only. The phone will operate in SIP mode.</p> | | |
| up.pictureFrame.timePerImage | 3 to 300 seconds | 5 |
| <p>VVX 500 and 1500 only. The number of seconds to display each picture frame image.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---------------------------|----------------|
| up.pictureFrame.folder | string | Null |
| <p>VVX 500 and 1500 only. The path name for images. The maximum length is 40 characters. If set to Null, images stored in the root folder on the USB flash drive are displayed. For example, if the images are stored in the <code>/images/phone</code> folder on the USB flash drive, set this parameter to <code>images/phone</code>.</p> | | |
| up.prioritizeBackgroundMenuItem.enable¹ | 0 or 1 | 1 |
| <p>If <code>up.idleBrowser.enabled</code> is 1, this parameter can be set to 1 to display a Prioritize Background menu to the user. The user can choose whether the phone background should take priority over the idle browser or not.</p> | | |
| up.screenCapture.enabled¹ | 0 or 1 | 0 |
| <p>If 0, screen captures are disabled. If 1, the user can enable screen captures from the Screen Capture menu on the phone. <i>Note:</i> when the phone reboots, screen captures are disabled from the Screen Capture menu on the phone.</p> | | |
| up.screenSaver.enabled | 0 or 1 | 0 |
| <p>VVX 500 and 1500 only. If 0, the screen saver feature is disabled. If 1, the 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 will cycle 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 <code>up.pictureFrame.folder</code>. The screen saver displays when the phone has been in the idle state for the amount of time specified by <code>up.screenSaver.waitTime</code>.</p> | | |
| up.screenSaver.waitTime | 1 to 9999, minutes | 15 |
| <p>VVX 500 and 1500 only. The number of minutes that the phone waits in the idle state before the screen saver starts.</p> | | |
| up.simplifiedSipCallInfo | 0 or 1 | 0 |
| <p>If 1, the displayed host name is trimmed for both incoming and outgoing calls and the protocol tag/information is not displayed for incoming and outgoing calls.</p> | | |
| up.toneControl.bass | -4 to 5, Null | 0 |
| <p>SoundPoint IP 7000 only. The bass equalization control. Each step is an increment or decrement of 1 dB at 225 kHz and 2 dB < 225 Hz.</p> | | |
| up.useDirectoryNames¹ | 0 or 1 | 1 |
| <p>If 0, names provided through network signaling are used for caller ID. If 1, the name field in the local contact directory will be used as the caller ID for incoming calls from contacts in the local directory. <i>Note:</i> Outgoing calls and corporate directory entries are not matched.</p> | | |
| up.warningLevel¹ | 0 to 2 | 0 |
| <p>If 0, the phone's warning icon and a pop-up message display on the phone for all warnings. If 1, the warning icon and pop-up messages are only shown for critical warnings. <i>Note:</i> All warnings are listed in the Warnings menu (navigate to Menu > Status > Diagnostics > Warnings on the phone).</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| up.welcomeSoundEnabled¹ | 0 or 1 | 1 |
| If 0, the welcome sound is disabled. If 1, the welcome sound is enabled and played each time the phone reboots. | | |
| up.welcomeSoundOnWarmBootEnabled¹ | 0 or 1 | 0 |
| If 0, the welcome sound is played when the phone powers up (cold boot), but not after it restarts or reboots (warm boot). If 1, the welcome sound plays each time the phone powers up, reboots, or restarts. | | |

¹ Change causes phone to restart or reboot.

<upgrade/>

You can specify the URL of a custom download server and the Polycom UC Software download server for the phone to check when searching for software upgrades.

Table 13-98: Upgrade Server Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|---|
| upgrade.custom.server.url | URL | Null |
| The URL of a custom download server. | | |
| upgrade.plcm.server.url | URL | http://downloads.polycom.com/voice/software/ |
| The URL of the Polycom UC Software download server. | | |

<video/>

This parameter is supported for use on the VVX 1500 only.

This parameter also includes:

- [<codecs/>](#)
- [<camera/>](#)
- [<localCameraView/>](#)

Table 13-99: General Video Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|----------------------------|----------------|
| video.autoFullScreen | 0 or 1 | 0 |
| If 0, video calls only use the full screen layout if it is explicitly selected by the user. If 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.autoStartVideoTx | 0 or 1 | 1 |
| When enabled, video transmission to the far side begins when you start a call. When disabled, video transmission does not begin until you press the Video > Start Video soft keys. This parameter controls video sent to the far side. Video from the far side will always be displayed if it is available, and far side users can control when to send video. | | |
| video.callMode.default | audio or video | audio |
| Allows the user to select the mode to use when using SIP protocol only. | | |
| video.callRate | 128 to 1024 | 512 |
| The default call rate (in kbps) to use when initially negotiating bandwidth for a video call. | | |
| video.dynamicControlMethod | 0 or 1 | 1 |
| If 1, the first I-Frame request uses the method defined by <code>video.forceRtcpVideoCodecControl</code> and subsequent requests alternate between RTCP-FB and SIP INFO. | | |
| video.enable | 0=Disable, 1=Enable | 1 |
| If 0, video is not enabled and all calls — both sent and received — are audio-only. If 1, video is sent in outgoing calls and received in incoming calls if the other device supports video. | | |
| video.forceRtcpVideoCodecControl¹ | 0 or 1 | 0 |
| If set to 1, the VVX 1500 is forced to send RTCP feedback messages to request fast update I-frames for all video calls (the phone includes <code>a=rtcp-fb</code> in the SDP. If 0, RTCP feedback messages are not forced. | | |
| video.iFrame.delay¹ | 0 to 10, seconds | 0 |
| When non-zero, an extra I-frame is transmitted after video starts. The amount of delay from the start of video until the I-frame is sent is configurable up to 10 seconds. Use a value of 2 seconds if you are using this parameter in a Microsoft Lync environment. | | |
| video.iFrame.minPeriod | 1 - 60 | 2 |
| After sending an I-frame, the phone will always wait at least this amount of time before sending another I-frame in response to requests from the far end. | | |
| video.iFrame.onPacketLoss | 0 or 1 | 0 |
| If 1, an I-frame is transmitted to the far end when a received RTCP report indicates that video RTP packet loss has occurred. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---------------------------|----------------|
| video.maxCallRate¹ | 128 to 1024 kbps | 768 |
| The maximum call rate allowed. This allows the administrator to limit the maximum call rate that the users can select. If <code>video.callRate</code> exceeds this value, this value will be used as the maximum. | | |
| video.quality¹ | motion, sharpness | Null |
| The optimal quality for video that you send in a call or a conference. Use <code>motion</code> if your outgoing video will have motion or movement. Use <code>sharpness</code> or <code>Null</code> if your outgoing video will have little or no movement. <i>Note:</i> If <code>motion</code> is not selected, moderate to heavy motion can cause some frames to be dropped. | | |
| video.screenMode | normal, full, crop | normal |
| The screen mode for the video window shown in non-full screen mode. If set to <code>normal</code> or <code>Null</code> , the entire view is displayed and horizontal or vertical black bars may appear on the edges to maintain the correct aspect ratio. If set to <code>full</code> , the entire view is stretched linearly and independently to fill the video frame. If set to <code>crop</code> , black bars are not shown, the image is re-sized and enlarged to cover the entire video frame, and parts of the image that do not fit in the display are cropped (removed). | | |
| video.screenModeFS | normal, full, crop | normal |
| The screen mode for the video window shown in full screen mode. If set to <code>normal</code> or <code>Null</code> , the entire view is displayed and horizontal or vertical black bars may appear on the edges to maintain the correct aspect ratio. If set to <code>full</code> , the entire view is stretched linearly and independently to fill the screen. If set to <code>crop</code> , black bars are not shown, the image is re-sized and enlarged to cover the entire screen, and parts of the image that do not fit in the display are cropped (removed). | | |

¹ Change causes phone to restart or reboot.

<codecs/>

These video codecs include:

- [<codecPref/>](#)
- [<profile/>](#)

<codecPref/>

This configuration parameter is defined as follows:

Table 13-100: Video Codec Preference Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| video.codecPref.H261¹ | 1 to 4 | 4 |
| video.codecPref.H264¹ | | 1 |
| video.codecPref.H2631998¹ | | 2 |
| video.codecPref.H263¹ | | 3 |

Specifies the video codec preferences for the VVX 1500 phone.

¹ Change causes phone to restart or reboot.

<profile/>

This section contains settings for a group of low-level video codec parameters. For most use cases, the default values will be appropriate. Polycom does not recommend changing the default values unless specifically advised to do so.

Table 13-101: Video Profile Parameters

| <i>Parameter</i> | <i>Permitted Values</i> |
|---|--|
| video.profile.H261.annexD¹ | 0 or 1 (default) |
| Enable or disable Annex D when negotiating video calls. | |
| video.profile.H261.CifMpi¹ | 1 (default) to 32 |
| Specify the frame rate divider that the phone uses when negotiating CIF resolution for a video call. You can enter a value between 0-4. To disable, enter '0'. The default frame rate divider is '1'. | |
| video.profile.H261.jitterBufferMax¹ | (video.profile.H261.jitter BufferMin + 500ms) to 2500ms, default 2000ms |
| The largest jitter buffer depth to be supported (in milliseconds). Jitter above this size will always cause lost packets. This parameter should be set to the smallest possible value that will support the expected network jitter. | |
| video.profile.H261.jitterBufferMin¹ | 33ms to 1000ms, default 150ms |
| The smallest jitter buffer depth (in milliseconds) that must be achieved before play out begins for the first time. Once this depth has been achieved initially, the depth may fall below this point and play out will still continue. This parameter should be set to the smallest possible value which is at least two packet payloads, and larger than the expected short term average jitter. | |
| video.profile.H261.jitterBufferShrink¹ | 33ms to 1000ms, default 70ms |
| The absolute minimum duration time (in milliseconds) of RTP packet Rx with no packet loss between jitter buffer size shrinks. Use smaller values (33 ms) to minimize the delay on known good networks. Use larger values (1000ms) to minimize packet loss on networks with large jitter (3000 ms). | |

| <i>Parameter</i> | <i>Permitted Values</i> |
|---|--|
| video.profile.H261.payloadType¹ | 0 to 127, default 31 |
| RTP payload format type for H261 MIME type. | |
| video.profile.H261.QcifMpi¹ | 1 (default) to 32 |
| Specify the frame rate divider that the phone uses when negotiating Quarter CIF resolution for a video call. You can enter a value between 0-4. To disable, enter '0'. The default frame rate divider is '1'. | |
| video.profile.H263.CifMpi¹ | 1 (default) to 32 |
| Specify the frame rate divider that the phone uses when negotiating CIF resolution for a video call. You can enter a value between 0-32. To disable, enter '0'. The default frame rate divider is '1'. | |
| video.profile.H263.jitterBufferMax¹ | (video.profile.H263.jitter BufferMin + 500ms) to 2500ms, default 2000ms |
| The largest jitter buffer depth to be supported (in milliseconds). Jitter above this size will always cause lost packets. This parameter should be set to the smallest possible value that will support the expected network jitter. | |
| video.profile.H263.jitterBufferMin¹ | 33ms to 1000ms, default 150ms |
| The smallest jitter buffer depth (in milliseconds) that must be achieved before play out begins for the first time. Once this depth has been achieved initially, the depth may fall below this point and play out will still continue. This parameter should be set to the smallest possible value which is at least two packet payloads, and larger than the expected short term average jitter. | |
| video.profile.H263.jitterBufferShrink¹ | 33ms to 1000ms, default 70ms |
| The absolute minimum duration time (in milliseconds) of RTP packet Rx with no packet loss between jitter buffer size shrinks. Use smaller values (33 ms) to minimize the delay on known good networks. Use larger values (1000ms) to minimize packet loss on networks with large jitter (3000 ms). | |
| video.profile.H263.payloadType¹ | 0 to 127, default 34 |
| RTP payload format type for H263 MIME type. | |
| video.profile.H263.QcifMpi¹ | 1 (default) to 32 |
| Specify the frame rate divider that the phone uses when negotiating Quarter CIF resolution for a video call. You can enter a value between 0-32. To disable, enter '0'. The default frame rate divider is '1'. | |
| video.profile.H263.SqcifMpi¹ | 1 (default) to 32 |
| Specify the frame rate divider that the phone uses when negotiating Sub Quarter CIF resolution for a video call. You can enter a value between 0-32. To disable, enter '0'. The default frame rate divider is '1'. | |
| video.profile.H2631998.annexF¹ | 0 (default) or 1 |
| Enable or disable Annex F when negotiating video calls. | |
| video.profile.H2631998.annexI¹ | 0 (default) or 1 |
| Enable or disable Annex I when negotiating video calls. | |

| <i>Parameter</i> | <i>Permitted Values</i> |
|---|--|
| video.profile.H2631998.annexJ¹ | 0 (default) or 1 |
| Enable or disable Annex J when negotiating video calls. | |
| video.profile.H2631998.annexK¹ | 0, 1 (default), 2, 3, 4 |
| Specify the value of Annex K to use when negotiating video calls. You can enter a value between 0-4. To disable, enter '0'. The default value is '1'. | |
| video.profile.H2631998.annexN¹ | 0, 1 (default), 2, 3, 4 |
| Specify the value of Annex N to use when negotiating video calls. You can enter a value between 0-4. To disable, enter '0'. The default value is '1'. | |
| video.profile.H2631998.annexT¹ | 0 (default) or 1 |
| Enable or disable Annex T when negotiating video calls. | |
| video.profile.H2631998.CifMpi¹ | 1 (default) to 32 |
| Specify the frame rate divider that the phone uses when negotiating CIF resolution for a video call. You can enter a value between 0-32. To disable, enter '0'. The default frame rate divider is '1'. | |
| video.profile.H2631998.jitterBufferMax¹ | (video.profile.H2631998.jitterBufferMin+ 500ms) to 2500ms, default 2000ms |
| The largest jitter buffer depth to be supported (in milliseconds). Jitter above this size will always cause lost packets. This parameter should be set to the smallest possible value that will support the expected network jitter. | |
| video.profile.H2631998.jitterBufferMin¹ | 33ms to 1000ms, default 150ms |
| The smallest jitter buffer depth (in milliseconds) that must be achieved before play out begins for the first time. Once this depth has been achieved initially, the depth may fall below this point and play out will still continue. This parameter should be set to the smallest possible value which is at least two packet payloads, and larger than the expected short term average jitter. | |
| video.profile.H2631998.jitterBufferShrink¹ | 33ms to 1000ms, default 70ms |
| The absolute minimum duration time (in milliseconds) of RTP packet Rx with no packet loss between jitter buffer size shrinks. Use smaller values (33 ms) to minimize the delay on known good networks. Use larger values (1000ms) to minimize packet loss on networks with large jitter (3000 ms). | |
| video.profile.H2631998.payloadType¹ | 96 (default) to 127 |
| RTP payload format type for H263-1998/90000 MIME type. | |
| video.profile.H2631998.QcifMpi¹ | 1 (default) to 32 |
| Specify the frame rate divider that the phone uses when negotiating Quarter CIF resolution for a video call. You can enter a value between 0-32. To disable, enter '0'. The default frame rate divider is '1'. | |
| video.profile.H2631998.SqcifMpi¹ | 1 (default) to 32 |
| Specify the frame rate divider that the phone uses when negotiating Sub Quarter CIF resolution for a video call. You can enter a value between 0-32. To disable, enter '0'. The default frame rate divider is '1'. | |

| <i>Parameter</i> | <i>Permitted Values</i> |
|---|--|
| video.profile.H264.jitterBufferMax¹ | (video.profile.H264.jitter BufferMin + 500ms) to 2500ms, default 2000ms |
| The largest jitter buffer depth to be supported (in milliseconds). Jitter above this size will always cause lost packets. This parameter should be set to the smallest possible value that will support the expected network jitter. | |
| video.profile.H264.jitterBufferMin¹ | 33ms to 1000ms, default 150ms |
| The smallest jitter buffer depth (in milliseconds) that must be achieved before play out begins for the first time. Once this depth has been achieved initially, the depth may fall below this point and play out will still continue. This parameter should be set to the smallest possible value which is at least two packet payloads, and larger than the expected short term average jitter. | |
| video.profile.H264.jitterBufferShrink¹ | 33ms to 1000ms, default 70ms |
| The absolute minimum duration time (in milliseconds) of RTP packet Rx with no packet loss between jitter buffer size shrinks. Use smaller values (33 ms) to minimize the delay on known good networks. Use larger values (1000ms) to minimize packet loss on networks with large jitter (3000 ms). | |
| video.profile.H264.payloadType¹ | 96 to 127, default 109 |
| RTP payload format type for H264/90000 MIME type. | |
| video.profile.H264.profileLevel¹ | 1, 1b, 1.1, 1.2, and 1.3 (default) |
| Specify the highest profile level within the Baseline profile supported in video calls. The phone supports the following levels: 1, 1b, 1.1, 1.2, 1.3. The default level is 1.3. For more information, refer to ITU-T H.264. | |

¹ Change causes phone to restart or reboot.

<camera/>

These settings control the performance of the camera. They are defined as follows:

Table 13-102: Video Camera Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| video.camera.brightness | 0 to 6 | 3 |
| Set brightness level. The value range is from 0 (Dimmest) to 6 (Brightest). | | |
| video.camera.contrast | 0 to 4 | 0 |
| Set contrast level.
The value range is from 0 (No contrast increase) to 3 (Most contrast increase), and 4 (Noise reduction contrast). | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| video.camera.flickerAvoidance | 0 to 2 | 0 |
| Set flicker avoidance.
If set to 0, flicker avoidance is automatic.
If set to 1, 50hz AC power frequency flicker avoidance (Europe/Asia).
If set to 2, 60hz AC power frequency flicker avoidance (North America). | | |
| video.camera.frameRate | 5 to 30 | 25 |
| Set target frame rate (frames per second). Values indicate a fixed frame rate, from 5 (least smooth) to 30 (most smooth).
<i>Note:</i> If <code>video.camera.frameRate</code> is set to a decimal number, the value 25 is used. | | |
| video.camera.saturation | 0 to 6 | 3 |
| Set saturation level. The value range is from 0 (Lowest) to 6 (Highest). | | |
| video.camera.sharpness | 0 to 6 | 3 |
| Set sharpness level. The value range is from 0 (Lowest) to 6 (Highest). | | |

<localCameraView/>

These settings control how the local camera is viewed on the screen. The configuration parameters are defined as follows:

Table 13-103: Local Camera View Preferences

| <i>Parameters</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|----------------------------|---------------------|
| video.localCameraView.fullscreen.enabled | 0=Disable, 1=Enable | 1 |
| Determines whether the local camera view is shown in the full screen layout.
If set to 0, the local camera view is not shown. If set to 1, the local camera view is shown. | | |
| video.localCameraView.fullscreen.mode | pip, side-by-side | side-by-side |
| Determines how the local camera view is shown. If set to pip, the local camera view displays as a picture-in-picture with the far end window.
If set to side-by-side, the local camera view displays side-by-side with the far end window. | | |

<voice/>

The <voice/> parameter controls the settings related to the audio on the phone.

Table 13-104: Voice Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--------------------------|----------------|
| voice.txPacketDelay¹ | low, normal, Null | Null |
| <p>If set to <i>normal</i> or Null, no audio parameters are changed.</p> <p>If set to <i>low</i> and there are no precedence conflicts, the following changes are made:</p> <ul style="list-style-type: none"> • <code>voice.codecPref.G722="1"</code> • <code>voice.codecPref.G711Mu="2"</code> • <code>voice.codecPref.G711A="3"</code> • <code>voice.codecPref.<OtherCodecs>=""</code> • <code>voice.audioProfile.G722.payloadSize="10"</code> • <code>voice.audioProfile.G711Mu.payloadSize="10"</code> • <code>voice.audioProfile.G711A.payloadSize="10"</code> • <code>voice.aec.hs.enable="0"</code> • <code>voice.ns.hs.enable="0"</code> | | |
| voice.txPacketFilter¹ | 0 or 1 | Null |
| <p>If 0, no Tx filtering is performed. If 1, narrowband Tx high pass filter is enabled.</p> | | |

¹ Change causes phone to restart or reboot.

This parameter includes:

- `<codecPref/>`
- `<volume/>`
- `<vad/>`
- `<quality monitoring/>`
- `<rxQoS/>`

`<codecPref/>`

As of Polycom UC Software 3.3.0, you can configure a simplified set of codec properties for all phone models to improve consistency and reduce workload on the phones.

If a particular phone does not support a codec, the phone will ignore that codec and continue to the codec next in the priority. For example, using the default values, the highest-priority codec on a SoundPoint IP 650 phone is G.722 since that model doesn't support Siren22, G.722, or Siren14.

For more information on codecs on particular phones and priorities, see [Audio Codecs](#).


Note: iLBC, G.729, and G.726QI Support

All SoundPoint IP and SoundStation IP phones *except* the SoundStation IP 5000 and the SoundStation Duo support both iLBC and G.729 if they are configured. The SoundStation IP 5000 and the SoundStation Duo phones support iLBC or G.729AB. If you enable iLBC on the SoundStation Duo, G.726QI (available for Multicast Group Paging and Push-to-Talk) is not supported.

Table 13-105: Voice Codec Preferences

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--------------------------------|-------------------------|----------------|
| voice.codecPref.G711_A | 0 to 27 | 7 |
| voice.codecPref.G711_Mu | | 6 |
| voice.codecPref.G719.32kbps | | 0 |
| voice.codecPref.G719.48kbps | | 0 |
| voice.codecPref.G719.64kbps | | 0 |
| voice.codecPref.G722 | | 4 |
| voice.codecPref.G7221.16kbps | | 0 |
| voice.codecPref.G7221.24kbps | | 0 |
| voice.codecPref.G7221.32kbps | | 5 |
| voice.codecPref.G7221_C.24kbps | | 0 |
| voice.codecPref.G7221_C.32kbps | | 0 |
| voice.codecPref.G7221_C.48kbps | | 2 |
| voice.codecPref.G729_AB | | 8 |
| voice.codecPref.iLBC.13_33kbps | | 0 |
| voice.codecPref.iLBC.15_2kbps | | 0 |
| voice.codecPref.Lin16.8ksps | | 0 |
| voice.codecPref.Lin16.16ksps | | 0 |
| voice.codecPref.Lin16.32ksps | | 0 |
| voice.codecPref.Lin16.44_1ksps | | 0 |
| voice.codecPref.Lin16.48ksps | | 0 |
| voice.codecPref.Siren14.24kbps | | 0 |
| voice.codecPref.Siren14.32kbps | | 0 |
| voice.codecPref.Siren14.48kbps | | 3 |
| voice.codecPref.Siren22.32kbps | | 0 |
| voice.codecPref.Siren22.48kbps | | 0 |
| voice.codecPref.Siren22.64kbps | | 0 |

The priority of the codec. If 0 or Null, the codec is disabled. A value of 1 is the highest priority. If a phone does not support a codec, it will treat the setting as if it were 0 and not offer or accept calls with that codec.

<volume/>

In some countries, regulations state that a phone's receiver volume must be reset to a nominal level for each new call. This is the phone's default behavior. Using this parameter, you can set the receiver volume to persist across calls each time a user makes changes to the default volume level.

Table 13-106: Volume Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| voice.volume.persist.bluetooth.headset¹ | 0 or 1 | 0 |
| If 0, the Bluetooth headset will not be used for every call. If 1, the Bluetooth headset will be used for all calls. | | |
| voice.volume.persist.handset¹ | 0 or 1 | 0 |
| If 0, the handset receive volume will automatically reset to a nominal level after each call. If 1, the volume for each call will be the same as the previous call. If set to 1, the handset receive volume will persist across calls.
If set to 0, the handset receive volume will be reset to nominal at the start of each call. | | |
| voice.volume.persist.headset¹ | 0 or 1 | 0 |
| If 0, the headset receive volume will automatically rest to a nominal level after each call. If 1, the volume for each call will be the same as the previous call. | | |
| voice.volume.persist.handsfree¹ | 0 or 1 | 1 |
| If 0, the speakerphone receive volume will automatically rest to a nominal level after each call. If 1, the volume for each call will be the same as the previous call. | | |
| voice.volume.persist.usb.handsfree¹ | 0 or 1 | 1 |
| If 0, the USB headset will not be used. If 1, the USB headset will be used in handsfree mode. | | |
| voice.volume.persist.usbHeadset¹ | 0 or 1 | 0 |
| If 0, the USB headset will not be used. If 1, the USB headset will be used. | | |

¹ Change causes phone to restart or reboot.

<vad/>

These settings control the performance of the voice activity detection (silence suppression) feature.

Table 13-107: Voice Activity Detection (VAD) Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-----------------------------|----------------|
| voice.vad.signalAnnexB¹ | 0 or 1 | 1 |
| If 0, there is no change to SDP. If 1, Annex B is used and a new line is added to SDP depending on the setting of <code>voice.vadEnable</code> . | | |
| <ul style="list-style-type: none"> If <code>voice.vadEnable</code> is set to 1, add parameter line <code>a=fmtp:18 annexb="yes"</code> below <code>a=rtpmap...</code> parameter line (where '18' could be replaced by another payload). If <code>voice.vadEnable</code> is set to 0, add parameter line <code>a=fmtp:18 annexb="no"</code> below <code>a=rtpmap...</code> parameter line (where '18' could be replaced by another payload). | | |
| voice.vadEnable¹ | 0 or 1 | 0 |
| If 0, voice activity detection (VAD) is disabled. If 1, VAD is enabled. | | |
| voice.vadThresh¹ | integer from 0 to 30 | 15 |
| The threshold for determining what is active voice and what is background noise in dB. Sounds louder than this value will be considered active voice, and sounds quieter than this threshold will be considered background noise. This does not apply to G.729AB codec operation which has its own built-in VAD function. | | |

¹ Change causes phone to restart or reboot.

<quality monitoring/>

The following table shows the Voice Quality Monitoring parameters.

Table 13-108: Voice Quality Monitoring Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| voice.qualityMonitoring.collector.alert.moslq.threshold.critical¹ | 0 to 40 | 0 |
| The threshold value of listening MOS score (MOS-LQ) that causes phone to send a critical alert quality report. Configure the desired MOS value multiplied by 10. If 0 or Null, critical alerts are not generated due to MOS-LQ. | | |
| For example, a configured value of 28 corresponds to the MOS score 2.8. | | |
| voice.qualityMonitoring.collector.alert.moslq.threshold.warning¹ | 0 to 40 | 0 |
| 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. If 0 or Null, warning alerts are not generated due to MOS-LQ. | | |
| For example, a configured value of 35 corresponds to the MOS score 3.5. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|----------------|
| voice.qualityMonitoring.collector.alert.delay.threshold.critical¹ | 0 to 2000 | 0 |
| Threshold value of one way delay (in ms) that causes phone to send a critical alert quality report. If 0 or Null, critical alerts are not generated due to one-way delay. One-way delay includes both network delay and end system delay. | | |
| voice.qualityMonitoring.collector.alert.delay.threshold.warning¹ | 0 to 2000 | 0 |
| Threshold value of one way delay (in ms) that causes phone to send a critical alert quality report. If 0 or Null, warning alerts are not generated due to one-way delay. One-way delay includes both network delay and end system delay. | | |
| voice.qualityMonitoring.collector.enable.periodic¹ | 0 or 1 | 0 |
| If 0, periodic quality reports are not generated. If 1, periodic quality reports are generated throughout a call. | | |
| voice.qualityMonitoring.collector.enable.session¹ | 0 or 1 | 0 |
| If 0, quality reports are not generated at the end of each call. If 1, reports are generated at the end of each call. | | |
| voice.qualityMonitoring.collector.enable.triggeredPeriodic¹ | 0 to 2 | 0 |
| If 0, alert states do not cause periodic reports to be generated. If 1, periodic reports are generated if an alert state is critical. If 2, period reports are generated when an alert state is either warning or critical.
<i>Note:</i> This parameter is ignored when <code>voice.qualityMonitoring.collector.enable.periodic</code> is 1, since reports are sent throughout the duration of a call. | | |
| voice.qualityMonitoring.collector.period¹ | 5 to 20 | 20 |
| The time interval between successive periodic quality reports. | | |
| voice.qualityMonitoring.collector.server.x.address¹
The server address | Dotted-decimal
IP address or
hostname | Null |
| voice.qualityMonitoring.collector.server.x.port¹
The server port. | 1 to 65535 | 5060 |
| The server address and 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. | | |
| voice.qualityMonitoring.rtcpxr.enable¹ | 0 or 1 | 0 |
| If 0, RTCP-XR packets are not generated. If 1, the packets are generated. | | |

¹ Change causes phone to restart or reboot.

<rxQoS/>

The following table lists the jitter buffer parameters for wired network interface voice traffic, wireless network interface voice traffic, and push-to-talk interface voice traffic.

Table 13-109: Voice Jitter Buffer Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| voice.rxQoS.avgJitter¹
The typical average jitter. | 0 to 80 | 20 |
| voice.rxQoS.maxJitter¹
The maximum expected jitter. | 0 to 200 | 160 |
| <p>The average and maximum jitter in milliseconds for wired network interface voice traffic.</p> <p><i>avgJitter</i> – The wired interface minimum depth will be automatically configured to adaptively handle this level of continuous jitter without packet loss.</p> <p><i>maxJitter</i> – The wired interface jitter buffer maximum depth will be automatically configured to handle this level of intermittent jitter without packet loss.</p> <p>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. Note that if legacy <i>voice.audioProfile.x.jitterBuffer.*</i> parameters are explicitly specified, they will be used to configure the jitter buffer and these <i>voice.rxQoS</i> parameters will be ignored.</p> | | |
| voice.rxQoS.wireless.avgJitter¹
The typical average jitter. | 0 to 200 | 70 |
| voice.rxQoS.wireless.maxJitter¹
The maximum expected jitter. | 20 to 500 | 300 |
| <p>The average and maximum jitter in milliseconds for wireless network interface voice traffic.</p> <p><i>avgJitter</i> – The wireless interface minimum depth will be automatically configured to adaptively handle this level of continuous jitter without packet loss.</p> <p><i>maxJitter</i> – The wireless interface jitter buffer maximum depth will be automatically configured to handle this level of intermittent jitter without packet loss.</p> <p>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.</p> <p><i>Note:</i> if legacy <i>voice.audioProfile.x.jitterBuffer.*</i> parameters are explicitly specified, they will be used to configure the jitter buffer and these <i>voice.rxQoS</i> parameters will be ignored for wireless interfaces.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| voice.rxQoS.ptt.avgJitter¹
The typical average jitter. | 0 to 200 | 150 |
| voice.rxQoS.ptt.maxJitter¹
The maximum expected jitter. | 20 to 500 | 480 |

The average and maximum jitter in milliseconds for IP multicast voice traffic (wired or wireless).

avgJitter – The PTT/Paging interface minimum depth will be automatically configured to adaptively handle this level of continuous jitter without packet loss.

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.

Note: if legacy `voice.audioProfile.x.jitterBuffer.*` parameters are explicitly specified, they will be used to configure the jitter buffer and these `voice.rxQoS` parameters will be ignored for PTT/Paging interface interfaces.

¹ Change causes phone to restart or reboot.

<volpProt/>

You must set up the call server and DTMF signaling parameters.

This parameter includes:

- <server/>
- <SDP/>
- <SIP/>
- <H323/>

<server/>

This configuration parameter is defined as follows:

Table 13-110: VoIP Server Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| volpProt.server.dhcp.available¹ | 0 or 1 | 0 |

If 0, do not check with the DHCP server for the SIP server IP address. If 1, check with the server for the IP address.

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---|----------------|
| volpProt.server.dhcp.option¹ | 128 to 254 | 128 |
| <p>The option to request from the DHCP server if <code>voIpProt.server.dhcp.available= 1</code>.
 <i>Note:</i> If <code>reg.x.server.y.address</code> is non-Null, it takes precedence even if the DHCP server is available.</p> | | |
| volpProt.server.dhcp.type¹ | 0 or 1 | 0 |
| <p>Type to request from the DHCP server if <code>voIpProt.server.dhcp.available</code> is set to 1. If this parameter is set to 0, IP request address. If set to 1, request string</p> | | |
| volpProt.server.x.address | dotted- decimal IP address or hostname | Null |
| <p>The IP address or hostname and port of a SIP server that accepts registrations. Multiple servers can be listed starting with <code>x=1</code> to 4 for fault tolerance.</p> | | |
| volpProt.server.x.port | 0, 1 to 65535 | 0 |
| <p>The port of the server that specifies registrations. If 0, the port used depends on <code>voIpProt.server.x.transport</code>.</p> | | |
| volpProt.server.x.registerRetry.baseTimeOut | 10 - 120 | 60 |
| <p>The base time period to wait before a registration retry. Used in conjunction with <code>voIpProt.server.x.registerRetry.maxTimeOut</code> to determine how long to wait. The algorithm is defined in RFC 5626.
 If both parameters <code>voIpProt.server.x.registerRetry.baseTimeOut</code> and <code>reg.x.server.y.registerRetry.baseTimeOut</code> are set, the value of <code>reg.x.server.y.registerRetry.baseTimeOut</code> takes precedence.</p> | | |
| volpProt.server.x.registerRetry.maxTimeOut | 60 - 1800 | 60 |
| <p>The maximum time period to wait before a registration retry. Used in conjunction with <code>voIpProt.server.x.registerRetry.maxTimeOut</code> to determine how long to wait. The algorithm is defined in RFC 5626.
 If both parameters <code>voIpProt.server.x.registerRetry.maxTimeOut</code> and <code>reg.x.server.y.registerRetry.maxTimeOut</code> are set, the value of <code>reg.x.server.y.registerRetry.maxTimeOut</code> takes precedence.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|---------------------|
| volpProt.server.x.transport | DNSnaptr,
TCPpreferred,
UDPOnly, TLS,
TCPOnly | DNSnaptr |
| <p>The transport method the phone uses to communicate with the SIP server.</p> <p>Null or DNSnaptr – 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.</p> <p>TCPpreferred – TCP is the preferred transport; UDP is used if TCP fails.</p> <p>UDPOnly: only UDP will be used.</p> <p>TLS – if TLS fails, transport fails. Leave port field empty (will default to 5061) or set to 5061.</p> <p>TCPOnly – only TCP will be used.</p> | | |
| volpProt.server.x.protocol.SIP | 0 or 1 | 1 |
| <p>If 1, server is a SIP proxy/registrar. <i>Note:</i> if set to 0, and the server is confirmed to be a SIP server, then the value is assumed to be 1.</p> | | |
| volpProt.server.x.expires | positive integer,
minimum 10 | 3600 |
| <p>The phone's requested registration period in seconds. <i>Note:</i> The period negotiated with the server may be different. The phone will attempt to re-register at the beginning of the overlap period. For example, if expires="300" and overlap="5", the phone will re-register after 295 seconds (300–5).</p> | | |
| volpProt.server.x.expires.overlap | 5 to 65535 | 60 |
| <p>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.</p> | | |
| volpProt.server.x.expires.lineSeize | positive integer,
minimum 0 was 10 | 30 |
| <p>Requested line-seize subscription period.</p> | | |
| volpProt.server.x.failOver.failBack.mode | newRequests,
DNSTTL,
registration,
duration | newRequest s |
| <p>The mode for failover failback:</p> <p>newRequests – all new requests are forwarded first to the primary server regardless of the last used server.</p> <p>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.</p> <p>registration – the phone tries the primary server again when the registration renewal signaling begins.</p> <p>duration – the phone tries the primary server again after the time specified by voIpProt.server.x.failOver.failBack.timeout.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| volpProt.server.x.failOver.failBack.timeout | 0, 60 to 65535 | 3600 |
| <p>If <code>voIpProt.server.x.failOver.failBack.mode</code> 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 will result in a timeout of 60 and 0 means do not fail-back until a fail-over event occurs with the current server.</p> | | |
| volpProt.server.x.failOver.failRegistrationOn | 0 or 1 | 0 |
| <p>When set to 1, and the <code>reRegisterOn</code> parameter is enabled, the phone will silently invalidate an existing registration (if it exists), at the point of failing over. When set to 0, and the <code>reRegisterOn</code> 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.</p> | | |
| volpProt.server.x.failOver.onlySignalWithRegistered | 0 or 1 | 1 |
| <p>When set to 1, and the <code>reRegisterOn</code> and <code>failRegistrationOn</code> 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 will end. No SIP messages will be sent to the unregistered server. When set to 0, and the <code>reRegisterOn</code> and <code>failRegistrationOn</code> parameters are enabled, 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).</p> | | |
| volpProt.server.x.failOver.reRegisterOn | 0 or 1 | 0 |
| <p>When set to 1, the phone will attempt 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 will proceed with the secondary server. When set to 0, the phone won't attempt to register with the second.</p> | | |
| volpProt.server.x.lcs | 0 or 1 | 0 |
| <p>If 0, the Microsoft Live Communications Server (LCS) is not supported. If 1, LCS is supported for registration x. This parameter overrides <code>voIpProt.SIP.lcs</code>.</p> | | |
| volpProt.server.x.register | 0 or 1 | 1 |
| <p>If 0, calls can be routed to an outbound proxy without registration. See <code>reg.x.server.y.register</code>. For more information, see Technical Bulletin 5844: SIP Server Fallback Enhancements on Polycom Phones.</p> | | |
| volpProt.server.x.retryTimeOut | 0 to 65535 | 0 |
| <p>The amount of time (in milliseconds) to wait between retries. If 0, use standard RFC 3261 signaling retry behavior.</p> | | |
| volpProt.server.x.retryMaxCount | 0 to 20 | 3 |
| <p>If set to 0, 3 is used. The number of retries that will be attempted before moving to the next available server.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---|-----------------|
| volpProt.server.x.specialInterop | standard, ocs2007r2, lcs2005, lync2010 | standard |
| Specify if this registration should support Microsoft Office Communications Server 2007 R2 (ocs2007r2), Microsoft Live Communications Server 2005 (lcs2005), or Microsoft Lync 2010 (lync2010). Note: For SpectraLink handsets, set this parameter to ocs2007r2 to use instant messaging. | | |
| volpProt.server.x.useOutboundProxy | 0 or 1 | 1 |
| Specify whether or not to use the outbound proxy specified in <code>voIpProt.SIP.outboundProxy.address</code> for server x. | | |
| volpProt.server.H323.x.address | dotted-decimal IP address or hostname | Null |
| Address of the H.323 gatekeeper. Note: Only one H.323 gatekeeper per phone is supported; if more than one is configured, only the first is used. | | |
| volpProt.server.H323.x.port | 0 to 65535 | 1719 |
| Port to be used for H.323 signaling.
<i>Note:</i> The H.323 gatekeeper RAS signaling uses UDP, while the H.225/245 signaling uses TCP. | | |
| volpProt.server.H323.x.expires | positive integer | 3600 |
| Desired registration period. | | |

¹ Change causes phone to restart or reboot.

<SDP/>

This configuration parameter is defined as follows:

Table 13-111: Session Description Protocol (SDP) Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| volpProt.SDP.answer.useLocalPreferences | 0 or 1 | 0 |
| If set to 1, the phones uses its own preference list when deciding which codec to use rather than the preference list in the offer. If set to 0, it is disabled.
<i>Note:</i> If the H.323 call from a Polycom VVX 1500 selects a lower-quality codec (H.261) but the called device also support H.264, this parameter should be enabled to resolve the situation. | | |
| volpProt.SDP.early.answerOrOffer | 0 or 1 | 0 |
| If set to 1, an SDP offer or answer is generated in a provisional reliable response and PRACK request and response. If set to 0, an SDP offer or answer is not generated.
<i>Note:</i> An SDP offer or answer is not generated if <code>reg.x.musicOnHold.uri</code> is set. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| volpProt.SDP.iLBC.13_33kbps.includeMode | 0 or 1 | 1 |
| <p>If set to 1, the phone should include the mode=30 FMTP parameter in SDP offers:</p> <ul style="list-style-type: none"> • If voice.codecPref.iLBC.13_33kbps is set and voice.codecPref.iLBC.15_2kbps is Null. • If voice.codecPref.iLBC.13_33kbps and voice.codecPref.iLBC.15_2kbps are both set, the iLBC 13.33 kbps codec is set to a higher preference. <p>If set to 0, the phone should not include the mode=30 FTMP parameter in SDP offers even if iLBC 13.33 kbps codec is being advertised. See <codecPref/>.</p> | | |
| volpProt.SDP.useLegacyPayloadTypeNegotiation | 0 or 1 | 0 |
| <p>If set to 1, the phone transmits and receives RTP using the payload type identified by the first codec listed in the SDP of the codec negotiation answer.</p> <p>If set to 0, RFC 3264 is followed for transmit and receive RTP payload type values.</p> | | |

<SIP/>

This configuration parameter is defined as follows:

Table 13-112: Session Initiation Protocol (SIP) Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|--|----------------|
| volpProt.SIP.acd.signalingMethod¹ | 0 or 1 | 0 |
| <p>If set to 0, the 'SIP-B' signaling is supported. (This is the older ACD functionality.)</p> <p>If set to 1, the feature synchronization signaling is supported. (This is the new ACD functionality.)</p> | | |
| volpProt.SIP.alertInfo.x.class | see the list of ring classes in <rt/> | default |
| <p>Alert-Info fields from INVITE requests will be compared against 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.</p> | | |
| volpProt.SIP.alertInfo.x.value | string | Null |
| <p>A string to match the alertinfo header in the incoming INVITE.</p> | | |
| volpProt.SIP.allowTransferOnProceeding | 0 to 1 | 1 |
| <p>If set to 1, a transfer can be completed during the proceeding state of a consultation call.</p> <p>If set to 0, a transfer is not allowed during the proceeding state of a consultation call.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---|----------------|
| volpProt.SIP.authOptimizedInFailover | 0 or 1 | 0 |
| <p>If set to 1, when failover occurs, the first new SIP request is sent to the server that sent the proxy authentication request.</p> <p>If set to 0, when failover occurs, the first new SIP request is sent to the server with the highest priority in the server list.</p> <p>If <code>reg.x.auth.optimizedInFailover</code> set to 0, this parameter is checked.</p> <p>If <code>voIpProt.SIP.authOptimizedInFailover</code> is 0, then this feature is disabled.</p> <p>If both parameters are set, the value of <code>reg.x.auth.optimizedInFailover</code> takes precedence.</p> | | |
| volpProt.SIP.CID.sourcePreference | ASCII string up to 120 characters long | Null |
| <p>Specify the priority order for the sources of caller ID information. The headers can be in any order. If Null, caller ID information comes from P-Asserted-Identity, Remote-Party-ID, and From in that order. The values <code>From,P-Asserted-Identity, Remote-Party-ID</code> and <code>P-Asserted-Identity,From, Remote-Party-ID</code> are also valid.</p> | | |
| volpProt.SIP.compliance.RFC3261.validate.contentLanguage | 0 or 1 | 1 |
| <p>If set to 1, validation of the SIP header content language is enabled. If set to 0, validation is disabled.</p> | | |
| volpProt.SIP.compliance.RFC3261.validate.contentLength | 0 or 1 | 1 |
| <p>If set to 1, validation of the SIP header content length is enabled.</p> | | |
| volpProt.SIP.compliance.RFC3261.validate.uriScheme | 0 or 1 | 1 |
| <p>If set to 1, validation of the SIP header URI scheme is enabled. If set to 0, validation is disabled.</p> | | |
| volpProt.SIP.conference.address | ASCII string up to 128 characters long | Null |
| <p>If Null, conferences are set up on the phone locally.</p> <p>If set to some value, conferences are set up by the server using the conferencing agent specified by this address. Acceptable values depend on the conferencing server implementation policy.</p> | | |
| volpProt.SIP.conference.parallelRefer | 0 or 1 | 0 |
| <p>If 1, a parallel REFER is sent to the call server. Note: <i>This parameter must be set for Siemens Openscape Centralized Conferencing.</i></p> | | |
| volpProt.SIP.connectionReuse.useAlias | 0 or 1 | 0 |
| <p>If set to 0, this is the old behavior.</p> <p>If set to 1, phone uses the connection reuse draft which introduces "alias".</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-----------------------------|----------------|
| volpProt.SIP.csta | 0 or 1 | 0 |
| If 0, the uaCSTA (User Agent Computer Supported Telecommunications Applications) feature is disabled. If 1, uaCSTA is enabled (If <code>reg.x.csta</code> is set, it will override this parameter). | | |
| volpProt.SIP.dialog.strictXLineID | 0 or 1 | 0 |
| If 0, the phone will not look for x-line-id (call appearance indec) in a SIP INVITE message, if one is not present. Instead, when it receives INVITE, the phone will generate the call appearance locally and pass that information to other parties involved in the call. | | |
| volpProt.SIP.dialog.usePvalue | 0 or 1 | 0 |
| If set to 0, phone uses a <code>pval</code> field name in Dialog. This obeys the draft-ietf-sipping-dialog-package-06.txt draft.
If set to 1, the phone uses a field name of <code>pvalue</code> . | | |
| volpProt.SIP.dialog.useSDP | 0 or 1 | 0 |
| If set to 0, a new dialog event package draft is used (no SDP in dialog body).
If set to 1, for backwards compatibility, use this setting to send SDP in the dialog body. | | |
| volpProt.SIP.dtmfViaSignaling.rfc2976¹ | 0 or 1 | 0 |
| If set to 1, DTMF digit information is sent in RFC2976 SIP INFO packets during a call.
If set to 0, no DTMF digit information is sent. | | |
| volpProt.SIP.enable¹ | 0 or 1 | 1 |
| A flag to determine if the SIP protocol is used for call routing, dial plan, DTMF, and URL dialing.
If set to 1, the SIP protocol is used.
Note: URL dialing is supported on SoundPoint IP 321/331/335 phones for unregistered lines only. | | |
| volpProt.SIP.failoverOn503Response | 0 or 1 | 1 |
| A flag to determine whether or not to trigger a failover if the phone receives a 503 response. | | |
| volpProt.SIP.header.diversion.enable¹ | 0 or 1 | 0 |
| If set to 1, the diversion header is displayed if received. If set to 0, the diversion header is not displayed. | | |
| volpProt.SIP.header.diversion.list.useFirst¹ | 0 or 1 | 1 |
| If set to 1, the first diversion header is displayed. If set to 0, the last diversion header is displayed. | | |
| volpProt.SIP.header.warning.codes.accept | comma separated list | Null |
| Specify a list of accepted warning codes.
If set to Null, all codes are accepted. Only codes between 300 and 399 are supported.
For example, if you want to accept only codes 325 to 330:
<code>voIpProt.SIP.header.warning.codes.accept=325,326,327,328,329,330</code>
Text will be shown in the appropriate language. For more information, see lcl_ml_lang_menu_x . | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| volpProt.SIP.header.warning.enable | 0 or 1 | 0 |
| If set to 1, the warning header is displayed if received. If set to 0, the warning header is not displayed. | | |
| volpProt.SIP.IM.autoAnswerDelay | 0 to 40, seconds | 10 |
| The time interval from receipt of the instant message invitation to automatically accepting the invitation. | | |
| volpProt.SIP.keepalive.sessionTimers | 0 or 1 | 0 |
| If set to 1, the session timer will be enabled. If set to 0, the session timer will be disabled, and the phone will not declare "timer" in "Support" header in an INVITE. The phone will still respond to a re-INVITE or UPDATE. The phone will not try to re-INVITE or UPDATE even if the remote end point asks for it. | | |
| volpProt.SIP.lcs | 0 or 1 | 0 |
| If 0, the Microsoft Live Communications Server (LCS) is not supported. If 1, LCS is supported. This parameter can set for a specific registration using <code>reg.x.lcs</code> . | | |
| volpProt.SIP.lineSeize.retries | 3 to 10 | 10 |
| Controls the number of times the phone will retry a notify when attempting to seize a line (BLA). | | |
| volpProt.SIP.local.port¹ | 0 to 65535 | 5060 |
| The local port for sending and receiving SIP signaling packets.
If set to 0, 5060 is used for the local port but is not advertised in the SIP signaling.
If set to some other value, that value is used for the local port and it is advertised in the SIP signaling. | | |
| volpProt.SIP.ms-forking | 0 or 1 | 0 |
| If set to 0, support for MS-forking is disabled. If set to 1, support for MS-forking is enabled and the phone will reject all Instant Message INVITEs. This parameter is applies when installing Microsoft Live Communications Server.
Note that if any end point registered to the same account has MS-forking disabled, all other end points default back to non-forking mode. Windows Messenger does not use MS-forking so be aware of this behavior if one of the end points is using Windows Messenger. | | |
| volpProt.SIP.mtls.enable | 0 or 1 | 1 |
| If 0, Mutual TLS is disabled. If 1, Mutual TLS is enabled. Used in conjunction with Microsoft Lync 2010. | | |
| volpProt.SIP.musicOnHold.uri | a SIP URI | Null |
| A URI that provides the media stream to play for the remote party on hold. This parameter is used if <code>reg.x.musicOnHold.uri</code> is Null.
<i>Note:</i> The SIP URI parameter transport is supported when configured with the values of UDP, TCP, or TLS. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|---|--------------------|
| volpProt.SIP.outboundProxy.address | dotted-decimal IP address or hostname | Null |
| The IP address or hostname of the SIP server to which the phone sends all requests. | | |
| volpProt.SIP.outboundProxy.port | 0 to 65535 | 0 |
| The port of the SIP server to which the phone sends all requests. | | |
| volpProt.SIP.outboundProxy.failOver.failBack.mode | newRequests, DNSTTL, registration, duration, | newRequests |
| <p>The mode for failover failback (overrides <code>voIpProt.server.x.failOver.failBack.mode</code>).</p> <p><code>newRequests</code> – all new requests are forwarded first to the primary server regardless of the last used server.</p> <p><code>DNSTTL</code> – 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.</p> <p><code>registration</code> – the phone tries the primary server again when the registration renewal signaling begins.</p> <p><code>duration</code> – the phone tries the primary server again after the time specified by <code>reg.x.outboundProxy.failOver.failBack.timeout</code> expires.</p> | | |
| volpProt.SIP.outboundProxy.failOver.failBack.timeout | 0, 60 to 65535 | 3600 |
| <p>The time to wait (in seconds) before failback occurs (overrides <code>voIpProt.server.x.failOver.failBack.timeout</code>). 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. If 0, the phone will not fail-back until a fail-over event occurs with the current server.</p> | | |
| volpProt.SIP.outboundProxy.failOver.failRegistrationOn | 0 or 1 | 0 |
| <p>When set to 1, and the <code>reRegisterOn</code> parameter is enabled, the phone will silently invalidate an existing registration (if it exists), at the point of failing over. When set to 0, and the <code>reRegisterOn</code> 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.</p> <p>Note that <code>voIpProt.SIP.outboundProxy.failOver.RegisterOn</code> must be enabled.</p> | | |
| volpProt.SIP.outboundProxy.failOver.onlySignalWithRegistered | 0 or 1 | 1 |
| <p>When set to 1, and the <code>reRegisterOn</code> and <code>failRegistrationOn</code> 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 will end. No SIP messages will be sent to the unregistered server. When set to 0, and the <code>reRegisterOn</code> and <code>failRegistrationOn</code> parameters are enabled, 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 <code>voIpProt.server.x.failOver.onlySignalWithRegistered</code>.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|--|-------------------|
| volpProt.SIP.outboundProxy.failOver.reRegisterOn | 0 or 1 | 0 |
| <p>This parameter overrides the <code>volpProt.server.x.failOver.reRegisterOn</code>. When set to <code>1</code>, the phone will attempt 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 will proceed with the secondary server. When set to <code>0</code>, 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.</p> | | |
| volpProt.SIP.outboundProxy.transport | DNSnaptr, TCPpreferred, UDPOnly, TLS, TCPOnly | DNSnaptr |
| <p>The transport method the phone uses to communicate with the SIP server.</p> <p>Null or DNSnaptr – if <code>reg.x.outboundProxy.address</code> is a hostname and <code>reg.x.outboundProxy.port</code> is 0 or Null, do NAPTR then SRV look-ups to try to discover the transport, ports and servers, as per RFC 3263. If <code>reg.x.outboundProxy.address</code> is an IP address, or a port is given, then UDP is used.</p> <p>TCPpreferred – TCP is the preferred transport, UDP is used if TCP fails.</p> <p>UDPOnly – only UDP will be used.</p> <p>TLS – if TLS fails, transport fails. Leave port field empty (will default to 5061) or set to 5061.</p> <p>TCPOnly – only TCP will be used.</p> | | |
| volpProt.SIP.pingInterval | 0 to 3600 | 0 |
| <p>The number in seconds to send "PING" message. This feature is disabled by default.</p> | | |
| volpProt.SIP.pingMethod | PING, OPTIONS | PING |
| <p>The ping method to be used.</p> | | |
| volpProt.SIP.presence.nortelShortMode¹ | 0 or 1 | 0 |
| <p>Different headers sent in SUBSCRIBE when used for presence on an Avaya (Nortel) server. Support is indicated by adding a header <code>Accept-Encoding: x-nortel-short</code>. A PUBLISH is sent to indicate the status of the phone.</p> | | |
| volpProt.SIP.requestValidation.digest.realm¹ | A valid string | PolycomSIP |
| <p>Determines the string used for Realm.</p> | | |
| volpProt.SIP.requestValidation.x.method¹ | Null, source, digest, both, all | Null |
| <p>If Null, no validation is made. Otherwise this sets the type of validation performed for the request: 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</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|---|----------------|
| volpProt.SIP.requestValidation.x.request¹ | INVITE, ACK ,
BYE,
REGISTER,
CANCEL,
OPTIONS,
INFO,
MESSAGE,
SUBSCRIBE,
NOTIFY,
REFER,
PRACK,
UPDATE | Null |
| Sets the name of the method for which validation will be applied.
<i>Note:</i> Intensive request validation may have a negative performance impact due to the additional signaling required in some cases. | | |
| volpProt.SIP.requestValidation.x.request.y.event¹ | A valid string | Null |
| Determines which events specified with the Event header should be validated; only applicable when <code>voIpProt.SIP.requestValidation.x.request</code> is set to <code>SUBSCRIBE</code> or <code>NOTIFY</code> .
If set to Null, all events will be validated. | | |
| volpProt.SIP.requestURI.E164.addGlobalPrefix | 0 or 1 | 0 |
| If set to 1, '+' global prefix is added to the E.164 user parts in sip: URIs. | | |
| volpProt.SIP.sendCompactHdrs | 0 or 1 | 0 |
| If set to 0, SIP header names generated by the phone use the long form, for example <code>From</code> .
If set to 1, SIP header names generated by the phone use the short form, for example <code>f</code> . | | |
| volpProt.SIP.serverFeatureControl.cf¹ | 0 or 1 | 0 |
| If set to 1, server-based call forwarding is enabled. The call server has control of call forwarding.
If set to 0, server-based call forwarding is not enabled. This is the old behavior. | | |
| volpProt.SIP.serverFeatureControl.dnd¹ | 0 or 1 | 0 |
| If set to 1, server-based DND is enabled. The call server has control of DND.
If set to 0, server-based DND is not enabled. This is the old behavior. | | |
| volpProt.SIP.serverFeatureControl.missedCalls¹ | 0 or 1 | 0 |
| If set to 1, server-based missed calls is enabled. The call server has control of missed calls.
If set to 0, server-based missed calls is not enabled. This is the old behavior. | | |
| volpProt.SIP.serverFeatureControl.localProcessing.cf | 0 or 1 | 1 |
| If set to 0 and <code>voIpProt.SIP.serverFeatureControl.cf</code> is set to 1, the phone will not perform local Call Forward behavior.
If set to 1, the phone will perform local Call Forward behavior on all calls received. | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| volpProt.SIP.serverFeatureControl.localProcessing.dnd | 0 or 1 | 1 |
| <p>If set to 0 and <code>voIpProt.SIP.serverFeatureControl.dnd</code> is set to 1, the phone will not perform local DND call behavior.</p> <p>If set to 1, the phone will perform local DND call behavior on all calls received.</p> | | |
| volpProt.SIP.specialEvent.checkSync.alwaysReboot¹ | 0 or 1 | 0 |
| <p>If set to 1, always reboot when a NOTIFY message is received from the server with event equal to check-sync.</p> <p>If set to 0, only reboot if any of the files listed in <MAC-address>.cfg have changed on the FTP server when a NOTIFY message is received from the server with event equal to check-sync.</p> | | |
| volpProt.SIP.specialEvent.lineSeize.nonStandard¹ | 0 or 1 | 1 |
| <p>If set to 1, process a 200 OK response for a line-seize event SUBSCRIBE as though a line-seize NOTIFY with Subscription State: active header had been received,. This speeds up processing.</p> | | |
| volpProt.SIP.strictLineSeize | 0 or 1 | 0 |
| <p>If set to 1, The phone is forced to wait for a 200 OK response when receiving a TRYING notify.</p> <p>If set to 0, this is old behavior.</p> | | |
| volpProt.SIP.strictReplacesHeader | 0 or 1 | 1 |
| <p>This parameter applies only to directed call pick-up attempts initiated against monitored BLF resources.</p> <p>If set to 1, the phone requires call-id, to-tag, and from-tag to perform a directed call-pickup when <code>call.directedCallPickupMethod</code> is configured as <code>native</code>.</p> <p>If set to 0, call pick-up requires a call id only.</p> | | |
| volpProt.SIP.strictUserValidation | 0 or 1 | 0 |
| <p>If set to 1, the phone is forced to match the user portion of signaling exactly.</p> <p>If set to 0, the phone will use the first registration if the user part does not match any registration.</p> | | |
| volpProt.SIP.tcpFastFailover | 0 or 1 | 0 |
| <p>If set to 1, failover occurs based on the values of <code>reg.x.server.y.retryMaxCount</code> and <code>voIpProt.server.x.retryTimeOut</code>.</p> <p>If set to 0, this is old behavior. See <code>reg.x.tcpFastFailover</code>.</p> | | |
| volpProt.SIP.tlsDsk.enable | 0 or 1 | 0 |
| <p>If 0, TLS DSK is disabled. If 1, TLS DSK is enabled. For more information, see Session Initiation Protocol (SIP) Authentication Extensions Protocol Overview.</p> | | |
| volpProt.SIP.turnOffNonSecureTransport¹ | 0 or 1 | 0 |
| <p>If set to 1, stop listening to port 5060 when using AS-SIP enabled.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------|
| volpProt.SIP.use486forReject | 0 or 1 | 0 |
| <p>If set to 1 and the phone is indicating a ringing inbound call appearance, the phone will transmit a 486 response to the received INVITE when the Reject soft key is pressed.</p> <p>If set to 0, no 486 response is transmitted.</p> | | |
| voipPort.SIP.useCompleteUriForRetrieve | 0 or 1 | 1 |
| <p>If set to 1, the target URI in BLF signaling will use the complete address as provided in the xml dialog document.</p> <p>If set to 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.</p> | | |
| volpProt.SIP.useContactInReferTo | 0 or 1 | 0 |
| <p>If set to 0, the "To URI" is used in the REFER.</p> <p>If set to 1, the "Contact URI" is used in the REFER.</p> | | |
| volpProt.SIP.useRFC2543hold | 0 or 1 | 0 |
| <p>If set to 0, use SDP media direction parameters (such as a=sendonly) per RFC 3264 when initiating a call. Otherwise use the obsolete c=0.0.0.0 RFC2543 technique. In either case, the phone processes incoming hold signaling in either format.</p> <p>Note: volpProt.SIP.useRFC2543hold is effective only when the call is initiated.</p> | | |
| volpProt.SIP.useSendonlyHold | 0 or 1 | 1 |
| <p>If set to 1, the phone will send a reinvite with a stream mode parameter of "sendonly" when a call is put on hold. This is the same as the previous behavior.</p> <p>If set to 0, the phone will send a reinvite with a stream mode parameter of "inactive" when a call is put on hold.</p> <p>NOTE: The phone will ignore the value of this parameter if set to 1 when the parameter volpProt.SIP.useRFC2543hold is also set to 1 (default is 0).</p> | | |

¹ Change causes phone to restart or reboot.

<H323/>

This parameter is used with the Polycom VVX 1500 phone only.

Table 13-113: H.323 Protocol Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| volpProt.H323.autoGateKeeperDiscovery¹ | 0 or 1 | 1 |
| <p>If set to 1, the phone will attempt to discover an H.323 gatekeeper address via the standard multicast technique, provided that a statically configured gatekeeper address is not available.</p> <p>If set to 0, the phone will not send out any gatekeeper discovery messages.</p> | | |

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|--|-------------------------|----------------|
| volpProt.H323.blockFacilityOnStartH245¹ | 0 or 1 | 0 |
| If set to 1, facility messages when using H.245 are removed. | | |
| volpProt.H323.dtmfViaSignaling.enabled¹ | 0 or 1 | 1 |
| If set to 1, the phone will use the H.323 signaling channel for DTMF key press transmission. | | |
| volpProt.H323.dtmfViaSignaling.H245alphanumericMode¹ | 0 or 1 | 1 |
| If set to 1, the phone will support H.245 signaling channel alphanumeric mode DTMF transmission.
<i>Note:</i> If both alphanumeric and signal modes can be used, the phone gives priority to DTMF. | | |
| volpProt.H323.dtmfViaSignaling.H245signalMode¹ | 0 or 1 | 1 |
| If set to 1, the phone will support H.245 signaling channel signal mode DTMF transmission. | | |
| volpProt.H323.enable¹ | 0 or 1 | 0 |
| A flag to determine if the H.323 protocol is used for call routing, dial plan, DTMF, and URL dialing.
If set to 1, the H.323 protocol is used. | | |
| volpProt.H323.local.port¹ | 0 to 65535 | 1720 |
| Local port to be used for H.323 signaling. Local port for sending and receiving H.323 signaling packets.
If set to 0, 1720 is used for the local port but is not advertised in the H.323 signaling.
If set to some other value, that value is used for the local port and it is advertised in the H.323 signaling. | | |
| volpProt.H323.local.RAS.port¹ | 1 to 65535 | 1719 |
| Local port for RAS signaling. | | |

¹ Change causes phone to restart or reboot.

<webutility/>

The webutility parameter is used to specify the download location of the translated language files for the Web Configuration Utility.

Table 13-114: Web Configuration Utility Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|---|
| webutility.languauge.plcm.server.url | URL | http://downloads.polycom.com/voice/software/languages/ |
| The download location of the translated language files for the Web Configuration Utility. | | |

<Wi-Fi/>

This parameter configures the use of the Ekahua Location System for the SpectraLink handsets.

Table 13-115: Wi-Fi Parameters

| <i>Parameter</i> | <i>Permitted Values</i> | <i>Default</i> |
|---|-------------------------|----------------------|
| wifi.rtls.ekahua.address | IP-address | 169.254.10.10 |
| The IP address of the Ekahau Positioning Engine. | | |
| wifi.rtls.ekahau.enable | 0 or 1 | 0 |
| If 0, the Ekahua Real-Time Location System (RTLS) is disabled. If 1, the Ekahua RTLS is enabled. | | |
| wifi.rtls.ekahua.port | 0 to 65535 | 8552 |
| The port number of the Ekahau Positioning Engine. | | |
| wifi.rtls.ekahua.txInterval | 0 to 2 | 0 |
| The maximum time between transmit intervals. If set to 0, the transmit interval is 1-minute. If set to 1, the transmit interval is 5-minutes. If set to 2, the transmit interval is 10-minutes. | | |

Chapter 14: Session Initiation Protocol (SIP)

This chapter describes the basic Session Initiation Protocol (SIP) and the protocol extensions that the current Polycom® UC Software supports.

This chapter contains information on:

- **Basic Protocols**—All the basic calling functionality described in the SIP specification is supported. Transfer is included in the basic SIP support.
- **Protocol Extensions**—Extensions add features to SIP that are applicable to a range of applications, including reliable 1xx responses and session timers.

For information on supported RFCs and Internet drafts, see the following section, [RFC and Internet Draft Support](#).

This chapter also describes:

- [Request Support](#)
- [Header Support](#)
- [Response Support](#)
- [Hold Implementation](#)
- [Reliability of Provisional Responses](#)
- [Transfer](#)
- [Third Party Call Control](#)
- [SIP for Instant Messaging and Presence Leveraging Extensions](#)
- [Shared Call Appearance Signaling](#)
- [Bridged Line Appearance Signaling](#)

RFC and Internet Draft Support

The following RFC's and Internet drafts are supported. For more information on any of the documents, enter the RFC number at <http://www.ietf.org/rfc.html>.

- RFC 1321—The MD5 Message-Digest Algorithm
- RFC 2327—SDP: Session Description Protocol
- RFC 2387—The MIME Multipart / Related Content-type
- RFC 2976—The SIP INFO Method
- RFC 3261—SIP: Session Initiation Protocol (replacement for RFC 2543)

- RFC 3262—Reliability of Provisional Responses in the Session Initiation Protocol (SIP)
- RFC 3263—Session Initiation Protocol (SIP): Locating SIP Servers
- RFC 3264—An Offer / Answer Model with the Session Description Protocol (SDP)
- RFC 3265—Session Initiation Protocol (SIP) - Specific Event Notification
- RFC 3311—The Session Initiation Protocol (SIP) UPDATE Method
- RFC 3325—SIP Asserted Identity
- RFC 3420—Internet Media Type message/sipfrag
- RFC 3515—The Session Initiation Protocol (SIP) Refer Method
- RFC 3555 — MIME Type of RTP Payload Formats
- RFC 3611 — RTP Control Protocol Extended reports (RTCP XR)
- RFC 3665—Session Initiation Protocol (SIP) Basic Call Flow Examples
- draft-ietf-sip-cc-transfer-05.txt—SIP Call Control - Transfer
- RFC 3725—Best Current Practices for Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP)
- RFC 3842—A Message Summary and Message Waiting Indication Event Package for the Session Initiation Protocol (SIP)
- RFC 3856—A Presence Event Package for Session Initiation Protocol (SIP)
- RFC 3891—The Session Initiation Protocol (SIP) “Replaces” Header
- RFC 3892—The Session Initiation Protocol (SIP) Referred-By Mechanism
- RFC 3959—The Early Session Disposition Type for the Session Initiation Protocol (SIP)
- RFC 3960—Early Media and Ringing Tone Generation in the Session Initiation Protocol (SIP)
- RFC 3968—The Internet Assigned Number Authority (IANA) Header Field Parameter Registry for the Session Initiation Protocol (SIP)
- RFC 3969—The Internet Assigned Number Authority (IANA) Uniform Resource Identifier (URI) Parameter Registry for the Session Initiation Protocol (SIP)
- RFC 4028—Session Timers in the Session Initiation Protocol (SIP)
- RFC 4235—An INVITE-Initiated Dialog Event Package for the Session Initiation Protocol (SIP)
- draft-levy-sip-diversion-08.txt—Diversion Indication in SIP
- draft-anil-sipping-bla-02.txt—Implementing Bridged Line Appearances (BLA) Using Session Initiation Protocol (SIP)
- draft-ietf-sip-privacy-04.txt—SIP Extensions for Network-Asserted Caller Identity and Privacy within Trusted Networks
- draft-ietf-sipping-cc-conferencing-03.txt—SIP Call Control - Conferencing for User Agents

- draft-ietf-sipping-rtcp-summary-02.txt —Session Initiation Protocol Package for Voice Quality Reporting Event
- draft-ietf-sip-connect-reuse-04.txt—Connection Reuse in the Session Initiation Protocol (SIP)

Request Support

The following SIP request messages are supported:

Table 14-1: Supported SIP Request Messages

| <i>Method</i> | <i>Supported</i> | <i>Notes</i> |
|---------------|------------------|---|
| REGISTER | Yes | |
| INVITE | Yes | |
| ACK | Yes | |
| CANCEL | Yes | |
| BYE | Yes | |
| OPTIONS | Yes | |
| SUBSCRIBE | Yes | |
| NOTIFY | Yes | |
| REFER | Yes | |
| PRACK | Yes | |
| INFO | Yes | RFC 2976, the phone does not generate INFO requests, but will issue a final response upon receipt. No INFO message bodies are parsed. |
| MESSAGE | Yes | Final response is sent upon receipt. Message bodies of type text/plain are sent and received. |
| UPDATE | Yes | |

Header Support

The following SIP request headers are supported:



Note: Reading the Following Tables

In the following table, a **Yes** in the Supported column means the header is sent and properly parsed.

Table 14-2: Supported SIP Request Headers

| <i>Header</i> | <i>Supported</i> |
|--------------------------|------------------|
| Accept | Yes |
| Accept-Encoding | Yes |
| Accept-Language | Yes |
| Accept-Resource-Priority | Yes |
| Access-Network-Info | No |
| Access-URL | Yes |
| Alert-Info | Yes |
| Allow | Yes |
| Allow-Events | Yes |
| Authentication-Info | Yes |
| Authorization | Yes |
| Call-ID | Yes |
| Call-Info | Yes |
| Contact | Yes |
| Content-Disposition | Yes |
| Content-Encoding | Yes |
| Content-Language | Yes |
| Content-Length | Yes |
| Content-Type | Yes |

| <i>Header</i> | <i>Supported</i> |
|-----------------------|---|
| CSeq | Yes |
| Date | Yes (for missed call, not used to adjust the time of the phone) |
| Diversion | Yes |
| Error-Info | No |
| Event | Yes |
| Expires | Yes |
| Flow-Timer | Yes |
| From | Yes |
| In-Reply-To | No |
| Join | Yes |
| Max-Forwards | Yes |
| Min-Expires | Yes |
| Min-SE | Yes |
| MIME-Version | No |
| Missed-Calls | Yes |
| ms-client-diagnostics | Yes |
| ms-keep-alive | Yes |
| ms-text-format | Yes |
| Organization | No |
| P-Asserted-Identity | Yes |
| P-Preferred-Identity | Yes |
| Priority | No |
| Privacy | No |
| Proxy-Authenticate | Yes |
| Proxy-Authorization | Yes |
| Proxy-Require | Yes |

| <i>Header</i> | <i>Supported</i> |
|--------------------|------------------|
| RAck | Yes |
| Reason | Yes |
| Record-Route | Yes |
| Refer-Sub | Yes |
| Refer-To | Yes |
| Referred-By | Yes |
| Referred-To | Yes |
| Remote-Party-ID | Yes |
| Replaces | Yes |
| Reply-To | No |
| Requested-By | No |
| Require | Yes |
| Resource-Priority | Yes |
| Response-Key | No |
| Retry-After | Yes |
| Route | Yes |
| RSeq | Yes |
| Server | Yes |
| Session-Expires | Yes |
| SIP-Etag | Yes |
| SIP-If-Match | Yes |
| Subject | Yes |
| Subscription-State | Yes |
| Supported | Yes |
| Timestamp | Yes |
| To | Yes |
| Unsupported | Yes |

| <i>Header</i> | <i>Supported</i> |
|---------------------|-------------------------------------|
| User-Agent | Yes |
| Via | Yes |
| voice-missed-call | Yes |
| Warning | Yes (Only warning codes 300 to 399) |
| WWW-Authenticate | Yes |
| X-Sipx-Authidentity | Yes |

Response Support

The following SIP responses are supported:



Note: Reading the Following Tables

In the following table, a **Yes** in the Supported column means the header is sent and properly parsed. The phone may not actually generate the response.

1xx Responses - Provisional

Table 14-3: Supported 1xx SIP Responses

| <i>Response</i> | <i>Supported</i> |
|-----------------------------|------------------|
| 100 Trying | Yes |
| 180 Ringing | Yes |
| 181 Call Is Being Forwarded | No |
| 182 Queued | No |
| 183 Session Progress | Yes |

2xx Responses - Success

Table 14-4: Supported 2xx SIP Responses

| <i>Response</i> | <i>Supported</i> | <i>Notes</i> |
|-----------------|------------------|--------------------|
| 200 OK | Yes | |
| 202 Accepted | Yes | In REFER transfer. |

3xx Responses - Redirection

Table 14-5: Supported 3xx SIP Responses

| <i>Response</i> | <i>Supported</i> |
|-------------------------|------------------|
| 300 Multiple Choices | Yes |
| 301 Moved Permanently | Yes |
| 302 Moved Temporarily | Yes |
| 305 Use Proxy | No |
| 380 Alternative Service | No |

4xx Responses - Request Failure



Note: Handling 4xx Responses

All 4xx responses for which the phone does not provide specific support will be treated the same as 400 Bad Request.

Table 14-6: Supported 4xx SIP Responses

| <i>Response</i> | <i>Supported</i> |
|----------------------|------------------|
| 400 Bad Request | Yes |
| 401 Unauthorized | Yes |
| 402 Payment Required | No |

| <i>Response</i> | <i>Supported</i> |
|-------------------------------------|------------------|
| 403 Forbidden | No |
| 404 Not Found | Yes |
| 405 Method Not Allowed | Yes |
| 406 Not Acceptable | No |
| 407 Proxy Authentication Required | Yes |
| 408 Request Timeout | No |
| 410 Gone | No |
| 413 Request Entity Too Large | No |
| 414 Request-URI Too Long | No |
| 415 Unsupported Media Type | Yes |
| 416 Unsupported URI Scheme | No |
| 420 Bad Extension | No |
| 421 Extension Required | No |
| 423 Interval Too Brief | Yes |
| 480 Temporarily Unavailable | Yes |
| 481 Call/Transaction Does Not Exist | Yes |
| 482 Loop Detected | Yes |
| 483 Too Many Hops | No |
| 484 Address Incomplete | Yes |
| 485 Ambiguous | No |
| 486 Busy Here | Yes |
| 487 Request Terminated | Yes |
| 488 Not Acceptable Here | Yes |
| 491 Request Pending | No |
| 493 Undecipherable | No |

5xx Responses - Server Failure

Table 14-7: Supported 5xx SIP Responses

| <i>Response</i> | <i>Supported</i> |
|---------------------------|------------------|
| 500 Server Internal Error | Yes |
| 501 Not Implemented | Yes |
| 502 Bad Gateway | No |
| 503 Service Unavailable | No |
| 504 Server Time-out | No |
| 505 Version Not Supported | No |
| 513 Message Too Large | No |

6xx Responses - Global Failure

Table 14-8: Supported 6xx SIP Responses

| <i>Response</i> | <i>Supported</i> |
|-----------------------------|------------------|
| 600 Busy Everywhere | No |
| 603 Decline | Yes |
| 604 Does Not Exist Anywhere | No |
| 606 Not Acceptable | No |

Hold Implementation

The phone supports two currently accepted means of signaling hold.

The first method, no longer recommended due in part to the RTCP problems associated with it, is to set the “c” destination addresses for the media streams in the SDP to zero, for example, c=0.0.0.0.

The second, and preferred, method is to signal the media directions with the “a” SDP media attributes sendonly, recvonly, inactive, or sendrecv. The hold signaling method used by the phone is configurable (see [SIP](#)), but both methods are supported when signaled by the remote end point

**Note: Hold Methods**

Even if the phone is set to use `c=0.0.0.0`, it will not do so if it gets any `sendrecv`, `sendonly`, or `inactive` from the server. These flags will cause it to revert to the other hold method.

Reliability of Provisional Responses

The phone fully supports RFC 3262 - *Reliability of Provisional Responses*.

Transfer

The phone supports transfer using the REFER method specified in draft-ietf-sip-cc-transfer-05 and RFC 3515.

Third Party Call Control

The phone supports the delayed media negotiations (INVITE without SDP) associated with third-party call-control applications.

When used with an appropriate server, the User Agent Computer Supported Telecommunications Applications (uaCSTA) feature on the phone may be used for remote control of the phone from computer applications such as Microsoft Office Communicator.

The phone is compliant with “Using CSTA for SIP Phone User Agents (uaCSTA), ECMA TR/087” for the Answer Call, Hold Call, and Retrieve Call functions and “Services for Computer Supported Telecommunications Applications Phase III, ECMA – 269” for the Conference Call function.

This feature is enabled by configuration parameters described in `<SIP/>` and `<reg/>` and needs to be activated by a feature application key.

SIP for Instant Messaging and Presence Leveraging Extensions

The phone is compatible with the Presence and Instant Messaging features of Microsoft Windows Messenger 5.1. In a future release, support for the Presence and Instant Message recommendations in the SIP Instant Messaging and Presence Leveraging Extensions (SIMPLE) proposals will be provided by the following Internet drafts or their successors:

- draft-ietf-simple-cpim-mapping-01
- draft-ietf-simple-presence-07
- draft-ietf-simple-presencelist-package-00
- draft-ietf-simple-winfo-format-02
- draft-ietf-simple-winfo-package-02

Shared Call Appearance Signaling

A shared line is an address of record managed by a call server. The server allows multiple end points to register locations against the address of record.

The phone supports shared call appearances (SCA) using the SUBSCRIBE-NOTIFY method in the “SIP Specific Event Notification” framework (RFC 3265). The events used are:

- “call-info” for call appearance state notification
- “line-seize for the phone to ask to seize the line

Bridged Line Appearance Signaling

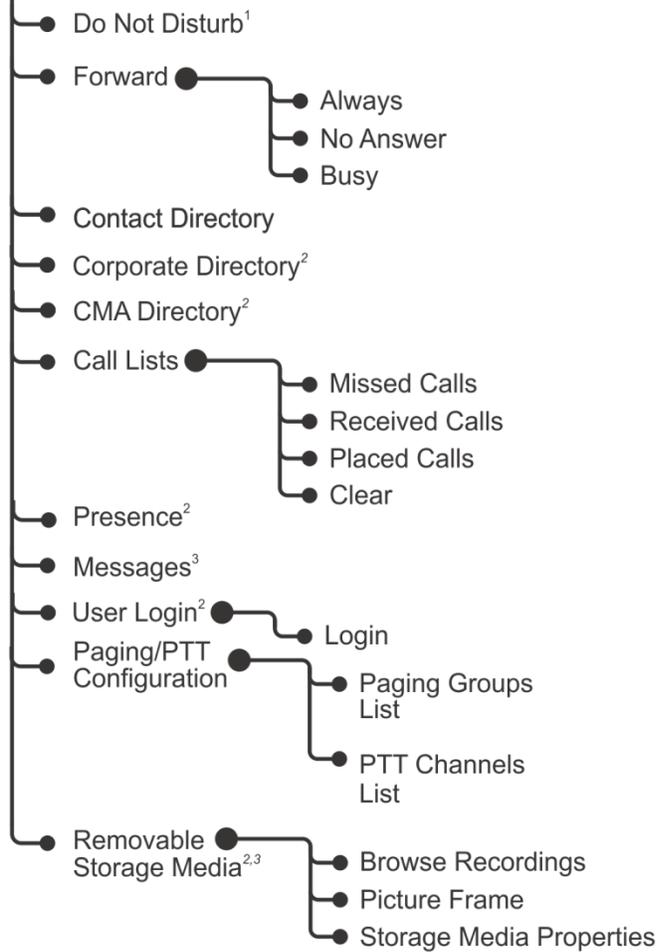
A bridged line is an address of record managed by a server. The server allows multiple end points to register locations against the address of record.

The phone supports bridged line appearances (BLA) using the SUBSCRIBE-NOTIFY method in the “SIP Specific Event Notification” framework (RFC 3265). The events used are:

- “dialog” for bridged line appearance subscribe and notify

Chapter 15: Polycom UC Software Menu System

Features

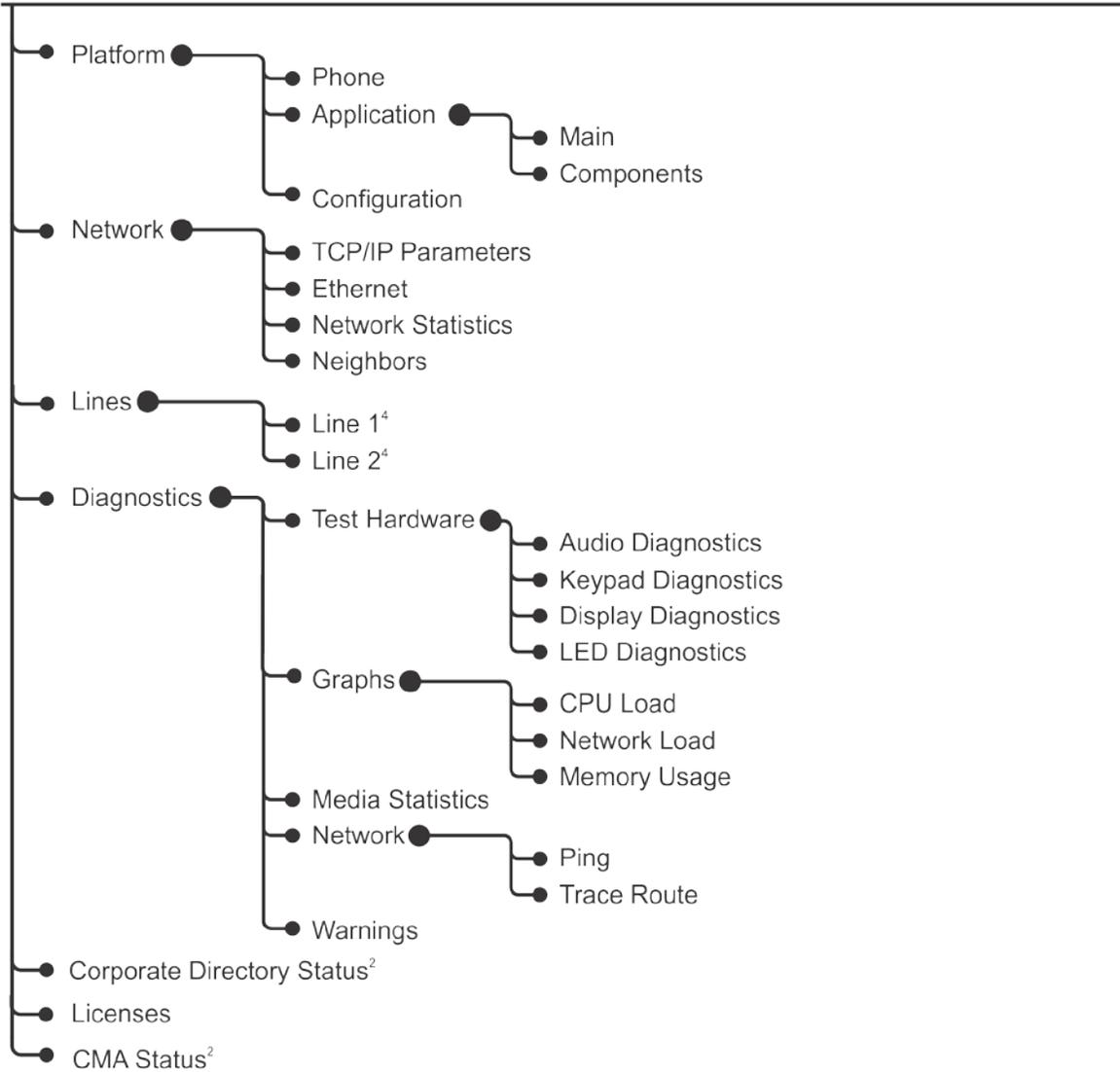


¹ If no hard key available.

² If enabled.

³ Platform dependent.

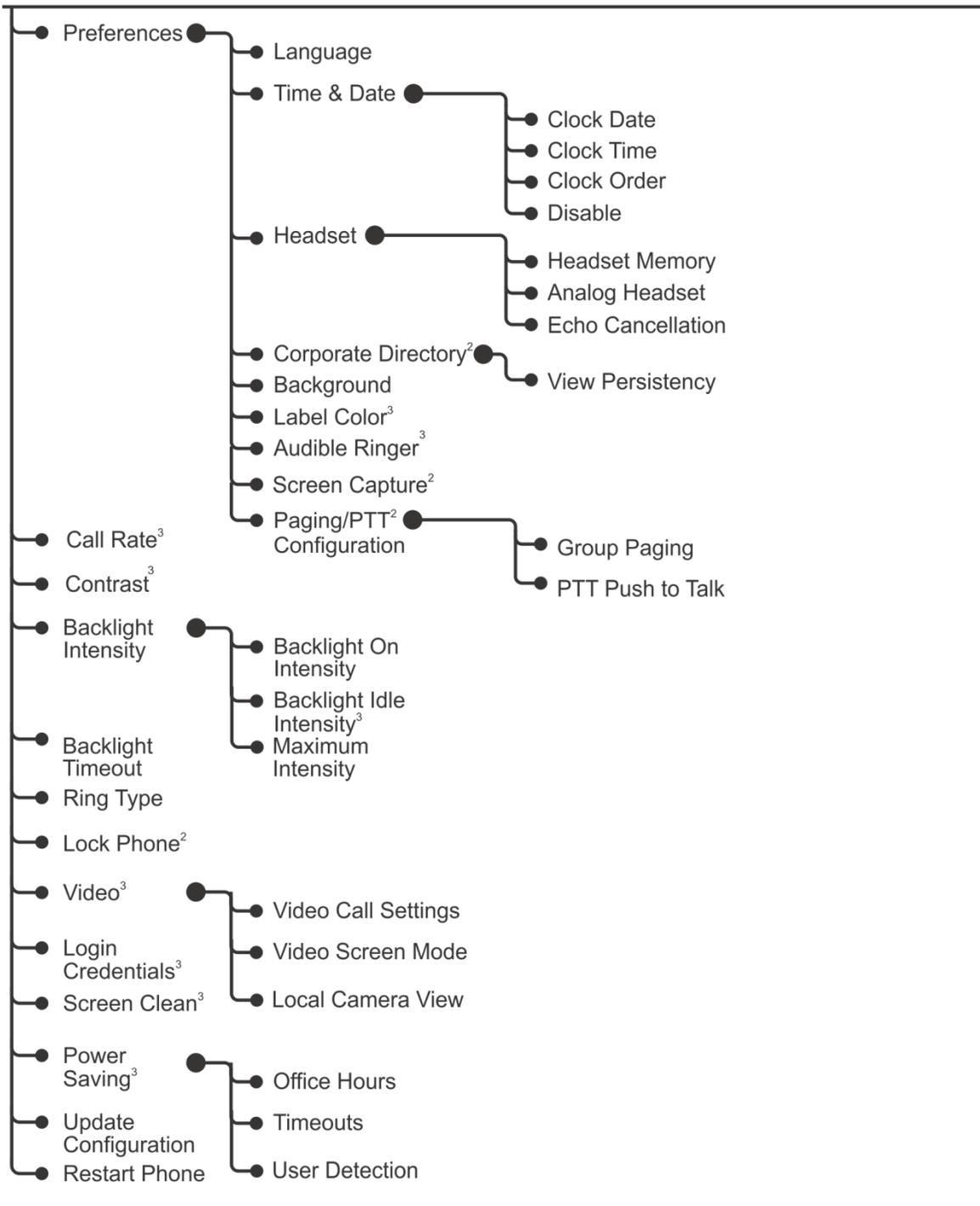
Status



² If enabled.

⁴ If applicable.

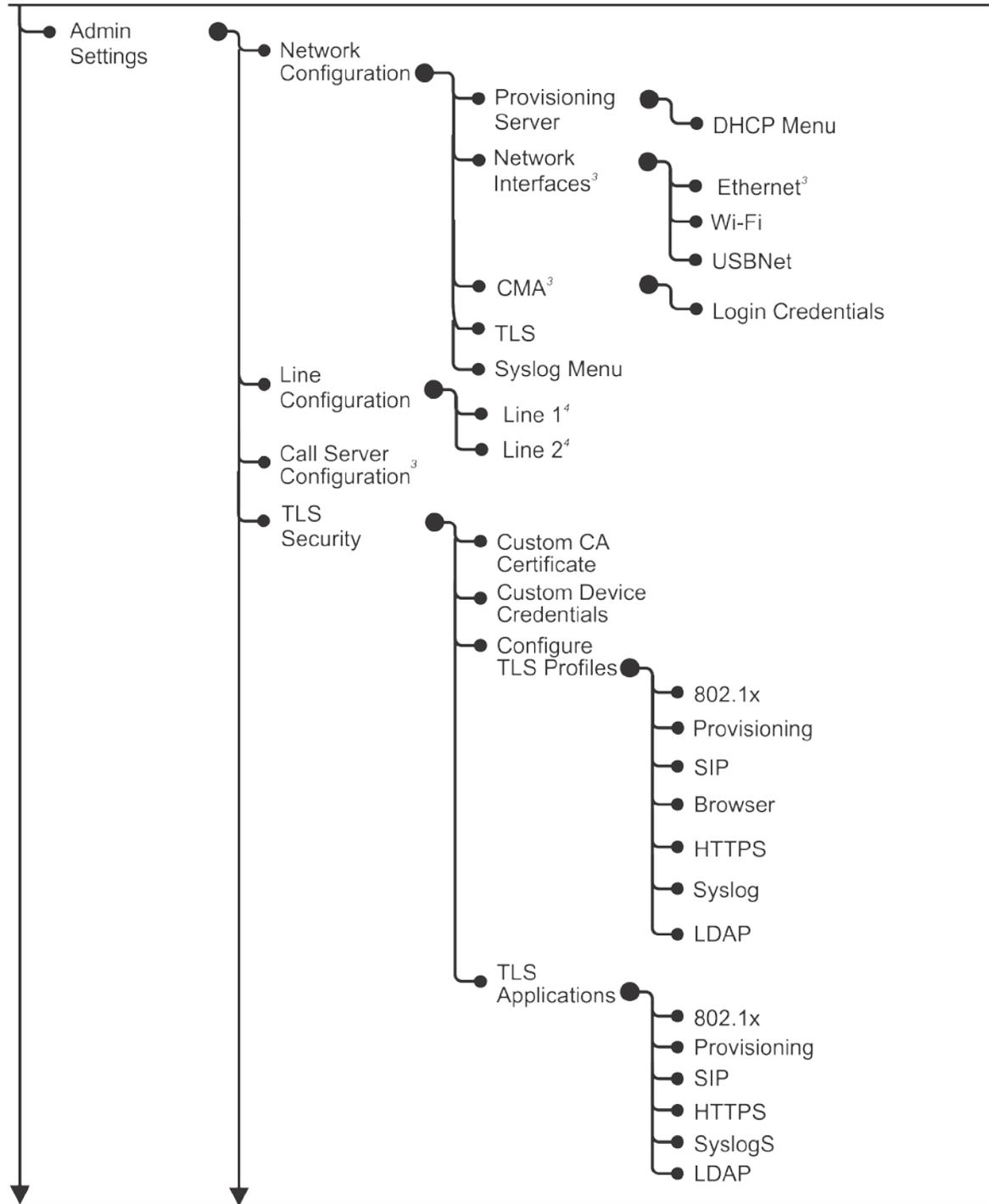
Settings > Basic



² If enabled.

³ Platform dependent.

Settings > Advanced⁵



³ Platform dependent.

⁴ If applicable.

⁵ Requires administrator password.

Directories⁶

- Contact Directory
- Corporate² Directory
- Call Lists ●
 - Missed Calls
 - Received Calls
 - Placed Calls
 - Clear

Messages⁶

Applications⁶

² If enabled.

⁶ Organization dependent.

Chapter 16: Third-Party Software

This chapter provides the copyright statements for third-party software products that have been incorporated into the Polycom® UC Software 4.1.0 application distribution.

Table 16-1: Third-Party Software

| <i>Product</i> | <i>License Location</i> |
|----------------|-------------------------|
| c-ares | c-ares |
| dhcp | dhcp 4.0.0-14 |
| droidfonts | droidfonts |
| Dropbear | Dropbear |
| eXpat | eXpat |
| freetype | freetype |
| gloox | gloox |
| ILG JPEG | IJG JPEG |
| libcurl | libcurl |
| libMng | libMng |
| liboil | liboil |
| libpcap | libpcap |
| libPng V2 | libpng V2 |
| libPng | libPng |
| libSRTP | libSRTP |
| libssh2 | libssh2 |
| ncurses | ncurses |
| OpenLDAP | OpenLDAP |
| OpenSSL | OpenSSL |

| <i>Product</i> | <i>License Location</i> |
|----------------|--|
| pmap | pmap-29092002 |
| winPcap | WinPcap |
| wpa_supplicant | wpa_supplicant |
| zlib | Table 12-8: SoundStation Duo Phone Key Functions |

Some Polycom products (specifically the VVX 500 and 1500 phones and SpectraLink handsets) may contain open source software that is licensed under the terms and conditions of the Free Software Foundation's GPL or LGPL licenses. See the [Polycom Voice OFFER of Source for GPL and LGPL Software](#).

Table 16-2: Open-Source Software

| <i>Product</i> | <i>Product</i> | <i>Product</i> |
|----------------|-------------------|----------------|
| alsa-lib | gst-openmax | mtd-utils |
| alsa-utils | gst-plugins-bad | procps |
| alsasink | gst-plugins-base | tsattach 1.0 |
| BlueZ | gst-plugins-good | tslib |
| BusyBox | gst-plugins-ugly | uboot |
| fbset | gststreamer | udev |
| ffmpeg | libsoup | Webkit |
| ffmpegdec | libomxil-bellagio | wireless-tools |
| freetype | libstdc++ | wrsv-ltt |
| glib2 | Linux kernel | x-loader |
| glibc | module-init-tools | |

c-ares

Copyright 1998 by the Massachusetts Institute of Technology.

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

dhcp 4.0.0-14

Copyright (c) 2004-2009 by Internet Systems Consortium, Inc. ("ISC")

Copyright (c) 1995-2003 by Internet Software Consortium

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND ISC DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL ISC BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Internet Systems Consortium, Inc.

950 Charter Street

Redwood City, CA 94063

<info@isc.org>

<http://www.isc.org/>

droidfonts

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work

by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and
2. You must cause any modified files to carry prominent notices stating that You changed the files; and
3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including

but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Dropbear

The majority of code is written by Matt Johnston, under the license below. Portions of the client-mode work are (c) 2004 Mihnea Stoenescu, under the same license:

Copyright (c) 2002-2006 Matt Johnston

Portions copyright (c) 2004 Mihnea Stoenescu

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====

LibTomCrypt and LibTomMath are written by Tom St Denis, and are Public Domain.

=====

sshpty.c is taken from OpenSSH 3.5p1,

Copyright (c) 1995 Tatu Ylonen <ylo@cs.hut.fi>, Espoo, Finland

All rights reserved

"As far as I am concerned, the code I have written for this software can be used freely for any purpose. Any derived versions of this software must be clearly marked as such, and if the derived work is incompatible with the protocol description in the RFC file, it must be called by a name other than "ssh" or "Secure Shell". "

=====

loginrec.c

loginrec.h

atomicio.h

atomicio.c

and strlcat() (included in util.c) are from OpenSSH 3.6.1p2, and are licensed under the 2 point BSD license.

loginrec is written primarily by Andre Lucas, atomicio.c by Theo de Raadt.

strlcat() is (c) Todd C. Miller

=====

Import code in keyimport.c is modified from PuTTY's import.c, licensed as follows:

PuTTY is copyright 1997-2003 Simon Tatham.

Portions copyright Robert de Bath, Joris van Rantwijk, Delian Delchev, Andreas Schultz, Jeroen Massar, Wez Furlong, Nicolas Barry, Justin Bradford, and CORE SDI S.A.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights

to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

eXpat

Copyright (c) 1998, 1999, 2000 Thai Open Source Software Center Ltd and Clark Cooper

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

freetype

The FreeType Project LICENSE

2006-Jan-27

Copyright 1996-2002, 2006 by

David Turner, Robert Wilhelm, and Werner Lemberg

Introduction

=====

The FreeType Project is distributed in several archive packages; some of them may contain, in addition to the FreeType font engine, various tools and contributions which rely on, or relate to, the FreeType Project.

This license applies to all files found in such packages, and which do not fall under their own explicit license. The license affects thus the FreeType font engine, the test programs, documentation and makefiles, at the very least.

This license was inspired by the BSD, Artistic, and IJG (Independent JPEG Group) licenses, which all encourage inclusion and use of free software in commercial and freeware products alike. As a consequence, its main points are that:

We don't promise that this software works. However, we will be interested in any kind of bug reports. ('as is' distribution)

You can use this software for whatever you want, in parts or full form, without having to pay us. ('royalty-free' usage)

You may not pretend that you wrote this software. If you use it, or only parts of it, in a program, you must acknowledge somewhere in your documentation that you have used the FreeType code. ('credits')

We specifically permit and encourage the inclusion of this software, with or without modifications, in commercial products.

We disclaim all warranties covering The FreeType Project and assume no liability related to The FreeType Project.

Finally, many people asked us for a preferred form for a credit/disclaimer to use in compliance with this license. We thus encourage you to use the following text:

Portions of this software are copyright © <year> The FreeType Project (www.freetype.org). All rights reserved.

gloox

Portions of this SOFTWARE PRODUCT are © 2006 by Jakob Schroeter <js@camaya.net>. All rights reserved.

IJG JPEG

Independent JPEG Group's free JPEG software

This package contains C software to implement JPEG image encoding, decoding, and transcoding. JPEG is a standardized compression method for full-color and gray-scale images.

The distributed programs provide conversion between JPEG "JFIF" format and image files in PBMPLUS PPM/PGM, GIF, BMP, and Targa file formats. The core compression and decompression library can easily be reused in other programs, such as image viewers. The package is highly portable C code; we have tested it on many machines ranging from PCs to Crays.

We are releasing this software for both noncommercial and commercial use. Companies are welcome to use it as the basis for JPEG-related products. We do not ask a royalty, although we do ask for an acknowledgement in product literature (see the README file in the distribution for details). We hope to make this software industrial-quality --- although, as with anything that's free, we offer no warranty and accept no liability.

For more information, contact jpeg-info@jpegclub.org.

Contents of this directory

`jpegsrc.vN.tar.gz` contains source code, documentation, and test files for release N in Unix format.

`jpegrN.zip` contains source code, documentation, and test files for release N in Windows format.

`jpegaltui.vN.tar.gz` contains source code for an alternate user interface for `cjpeg/djpeg` in Unix format.

`jpegaltuiN.zip` contains source code for an alternate user interface for `cjpeg/djpeg` in Windows format.

`wallace.ps.gz` is a PostScript file of Greg Wallace's introductory article about JPEG. This is an update of the article that appeared in the April 1991 Communications of the ACM.

`jpeg.documents.gz` tells where to obtain the JPEG standard and documents about JPEG-related file formats.

`jfif.ps.gz` is a PostScript file of the JFIF (JPEG File Interchange Format) format specification.

`jfif.txt.gz` is a plain text transcription of the JFIF specification; it's missing a figure, so use the PostScript version if you can.

`TIFFTechNote2.txt.gz` is a draft of the proposed revisions to TIFF 6.0's JPEG support.

`pm.errata.gz` is the errata list for the first printing of the textbook "JPEG Still Image Data Compression Standard" by Pennebaker and Mitchell.

`jdosaobj.zip` contains pre-assembled object files for `JMEMDOS.ASM`.

If you want to compile the IJG code for MS-DOS, but don't have an assembler, these files may be helpful.

libcurl

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2008, Daniel Stenberg, <daniel@haxx.se>.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

libMng

COPYRIGHT NOTICE:

Copyright © 2000-2008 Gerard Juyn (gerard@libmng.com)

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Gerard Juyn

The MNG Library is supplied "AS IS". The Contributing Authors disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the MNG Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.
2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors specifically permit, without fee, and encourage the use of this source code as a component to supporting the MNG and JNG file format in commercial products. If you use this source code in a product, acknowledgment would be highly appreciated.

LIBOIL

Copyright 2002,2003,2004,2005 David A. Schleef <ds@schleef.org>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The source code in the liboil/motovec directory is subject to the following license:

Copyright Motorola, Inc. 2003

ALL RIGHTS RESERVED

You are hereby granted a copyright license to use, modify, and distribute the SOFTWARE so long as this entire notice is retained without alteration in any modified and/or redistributed versions, and that such modified versions are clearly identified as such.

No licenses are granted by implication, estoppel or otherwise under any patents or trademarks of Motorola, Inc. The SOFTWARE is provided on an "AS IS" basis and without warranty.

To the maximum extent permitted by applicable law, MOTOROLA DISCLAIMS ALL WARRANTIES WHETHER EXPRESS OR IMPLIED, INCLUDING IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY AGAINST INFRINGEMENT WITH REGARD TO THE SOFTWARE (INCLUDING ANY MODIFIED VERSIONS THEREOF) AND ANY ACCOMPANYING WRITTEN MATERIALS.

To the maximum extent permitted by applicable law, IN NO EVENT SHALL MOTOROLA BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR OTHER PECUNIARY LOSS) ARISING OF THE USE OR INABILITY TO USE THE SOFTWARE. Motorola assumes no responsibility for the maintenance and support of the SOFTWARE.

The source code implementing the Mersenne Twister algorithm is subject to the following license:

Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

libpcap

Portions Copyright (c) 1994, 1995, 1996, 1997, 1998

The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: This product includes software developed by the Computer Systems Engineering Group at Lawrence Berkeley Laboratory.
4. Neither the name of the University nor of the Laboratory may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 1997 Yen Yen Lim and North Dakota State University

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: This product includes software developed by Yen Yen Lim and North Dakota State University
4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN

ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 2002 - 2005 NetGroup, Politecnico di Torino (Italy)

Copyright (c) 2005 - 2009 CACE Technologies, Inc. Davis (California)

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the Politecnico di Torino nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 2007 Fulko Hew, SITA INC Canada, Inc <fulko.hew@sita.aero>

License: BSD

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Portions Copyright (c) 2002 - 2005 NetGroup, Politecnico di Torino (Italy)

Copyright (c) 2005 - 2009 CACE Technologies, Inc. Davis (California)

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the Politecnico di Torino nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 2001 Atsushi Onoe

Copyright (c) 2002-2005 Sam Leffler, Errno Consulting

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

Alternatively, this software may be distributed under the terms of the GNU General Public License ("GPL") version 2 as published by the Free Software Foundation.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 1996

Juniper Networks, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that: (1) source code distributions retain the above copyright notice and this paragraph in its entirety, (2) distributions including binary code include the above copyright notice and this paragraph in its entirety in the documentation or other materials provided with the distribution. The name of Juniper Networks may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Portions Copyright (c) 2006 Paolo Abeni (Italy)

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 2000 Torsten Landschoff <torsten@debian.org>

Sebastian Kraemer <kraemer@cs.uni-potsdam.de>

License: BSD

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Modifications: Added PACKET_MMAP support

Paolo Abeni <paolo.abeni@email.it>

based on previous works of:

Simon Patarin <patarin@cs.unibo.it>

Phil Wood <cpw@lanl.gov>

Monitor-mode support for mac80211 includes code taken from the iw command; the copyright notice for that code is

Copyright (c) 2007, 2008 Johannes Berg

Copyright (c) 2007 Andy Lutomirski

Copyright (c) 2007 Mike Kershaw

Copyright (c) 2008 GÅ;bor Stefanik

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright 1989 by Carnegie Mellon.

Permission to use, copy, modify, and distribute this program for any purpose and without fee is hereby granted, provided that this copyright and permission notice appear on all copies and supporting documentation, the name of Carnegie Mellon not be used in advertising or publicity pertaining to distribution of the program without specific prior permission, and notice be given in supporting documentation that copying and distribution is by permission of Carnegie Mellon and Stanford University. Carnegie Mellon makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

libpng V2

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:

If you modify libpng you may insert additional notices immediately following this sentence.

libpng versions 1.2.6, August 15, 2004, through 1.2.24, December 14, 2007, are Copyright (c) 2004, 2006-2007 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.2.5 with the following individual added to the list of Contributing Authors:

Cosmin Truta

libpng versions 1.0.7, July 1, 2000, through 1.2.5, October 3, 2002, are Copyright (c) 2000-2002 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux

Eric S. Raymond

Gilles Vollant

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998, 1999, 2000 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

Tom Lane

Glenn Randers-Pehrson

Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996, 1997 Andreas Dilger

Distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

John Bowler

Kevin Bracey

Sam Bushell

Magnus Holmgren

Greg Roelofs

Tom Tanner

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger

Dave Martindale

Guy Eric Schalnat

Paul Schmidt

Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.
2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

libPng

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:

If you modify libpng you may insert additional notices immediately following this sentence.

This code is released under the libpng license.

libpng versions 1.2.6, August 15, 2004, through 1.2.40, September 10, 2009, are Copyright (c) 2004, 2006-2009 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.2.5 with the following individual added to the list of Contributing Authors

Cosmin Truta

libpng versions 1.0.7, July 1, 2000, through 1.2.5 - October 3, 2002, are Copyright (c) 2000-2002 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors

Simon-Pierre Cadieux

Eric S. Raymond

Gilles Vollant

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998, 1999 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

Tom Lane

Glenn Randers-Pehrson

Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996, 1997 Andreas Dilger Distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

John Bowler

Kevin Brace

Sam Bushell

Magnus Holmgren

Greg Roelofs

Tom Tanner

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger

Dave Martindale

Guy Eric Schalnat

Paul Schmidt

Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.
2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

Libpng is OSI Certified Open Source Software. OSI Certified Open Source is a certification mark of the Open Source Initiative.

Glenn Randers-Pehrson

glennrp at users.sourceforge.net

September 10, 2009

libSRTP

Copyright (c) 2001-2005 Cisco Systems, Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- * Neither the name of the Cisco Systems, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

libssh2

Copyright (c) 2004-2007 Sara Golemon <sarag@libssh2.org>

Copyright (C) 2006-2007 The Written Word, Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of the copyright holder nor the names of any other contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

ncurses

Copyright (c) 1998-2004, 2006 Free Software Foundation, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, distribute with modifications, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished - to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE ABOVE COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name(s) of the above copyright holders shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization.

OpenLDAP

The OpenLDAP Public License

Version 2.8, 17 August 2003

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

- 1: Redistributions in source form must retain copyright statements and notices,
- 2: Redistributions in binary form must reproduce applicable copyright statements and notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution, and
- 3: Redistributions must contain a verbatim copy of this document.

The OpenLDAP Foundation may revise this license from time to time.

Each revision is distinguished by a version number. You may use this Software under terms of this license revision or under the terms of any subsequent revision of the license.

THIS SOFTWARE IS PROVIDED BY THE OPENLDAP FOUNDATION AND ITS CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OPENLDAP FOUNDATION, ITS CONTRIBUTORS, OR THE AUTHOR(S) OR OWNER(S) OF THE SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The names of the authors and copyright holders must not be used in advertising or otherwise to promote the sale, use or other dealing in this Software without specific, written prior permission. Title to copyright in this Software shall at all times remain with copyright holders.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

Copyright 1999-2003 The OpenLDAP Foundation, Redwood City, California, USA. All Rights Reserved. Permission to copy and distribute verbatim copies of this document is granted.

OpenSSL

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License

Copyright (c) 1998-2008 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:

"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit
(<http://www.openssl.org/>)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License:

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)

All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com).

The implementation was written so as to conform with Netscape's SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used.

This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

"This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)"

The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related.

4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
- THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publicly available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

pmap-29092002

Copyright (c) 2002 Andrew Isaacson <adi@hexapodia.org>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

WinPcap

Copyright (c) 1999 - 2005 NetGroup, Politecnico di Torino (Italy).

Copyright (c) 2005 - 2010 CACE Technologies, Davis (California).

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the Politecnico di Torino, CACE Technologies nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes software developed by the University of California, Lawrence Berkeley Laboratory and its contributors.

This product includes software developed by the Kungliga Tekniska Högskolan and its contributors.

This product includes software developed by Yen Yen Lim and North Dakota State University.

Portions Copyright (c) 1990, 1991, 1992, 1993, 1994, 1995, 1996, 1997 The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes software developed by the University of California, Berkeley and its contributors."
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 1983 Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by the University of California, Berkeley. The name of the University may not be used to endorse or promote products derived from this software without specific prior written permission. THIS SOFTWARE IS PROVIDED ``AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Portions Copyright (c) 1995, 1996, 1997 Kungliga Tekniska Högskolan (Royal Institute of Technology, Stockholm, Sweden). All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes software developed by the Kungliga Tekniska Högskolan and its contributors."
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 1997 Yen Yen Lim and North Dakota State University. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes software developed by Yen Yen Lim and North Dakota State University"
4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 1993 by Digital Equipment Corporation.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies, and that the name of Digital Equipment Corporation not be used in advertising or publicity pertaining to distribution of the document or software without specific, written prior permission.

THE SOFTWARE IS PROVIDED "AS IS" AND DIGITAL EQUIPMENT CORP. DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL DIGITAL EQUIPMENT CORPORATION BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF

USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Portions Copyright (C) 1995, 1996, 1997, 1998, and 1999 WIDE Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE PROJECT AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE

ARE DISCLAIMED. IN NO EVENT SHALL THE PROJECT OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS

OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY

OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright (c) 1996 Juniper Networks, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that: (1) source code distributions retain the above copyright notice and this paragraph in its entirety, (2) distributions including binary code include the above copyright notice and this paragraph in its entirety in the documentation or other materials provided with the distribution. The name of Juniper Networks may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Portions Copyright (c) 2001 Daniel Hartmeier All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTOR "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Portions Copyright 1989 by Carnegie Mellon.

Permission to use, copy, modify, and distribute this program for any purpose and without fee is hereby granted, provided that this copyright and permission notice appear on all copies and supporting documentation, the name of Carnegie Mellon not be used in advertising or publicity pertaining to distribution of the program without specific prior permission, and notice be given in supporting documentation that copying and distribution is by permission of Carnegie Mellon and Stanford University. Carnegie Mellon makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

wpa_suppl

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name(s) of the above-listed copyright holder(s) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

zlib

version 1.2.3, July 18th, 2005

Copyright (C) 1995-2005 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly

Mark Adler

jloup@gzip.org

madler@alumni.caltech.edu

POLYCOM, INC.
APPLICATION PROGRAMMING INTERFACE LICENSE (“API”)
FOR SOUNDPOINT IP AND SOUNDSTATION IP PRODUCTS (“Product” or “Products”).

1. **Agreement.** You understand and agree that by using the API you will be bound by the terms of the End User License and Warranty Terms included with the Product(s) and this document (together, the “Agreement”). In the event of any conflicts between the End User License and Warranty Terms and this document, this document shall govern with respect to the API.
2. **Parties.** For purposes of this Agreement “you” or “your” shall mean the individual or entity accepting this Agreement or using the API. The relationship between you and Polycom is that of licensee/licensor. No legal partnership or agency relationship is created between you and Polycom. Neither you nor Polycom is a partner, an agent or has any authority to bind the other. You agree not to represent otherwise.
3. **License/Ownership.** Subject to your compliance with this Agreement, Polycom hereby grants you a limited license to use the API solely for the purposes of developing and testing your own proprietary software to be used in conjunction with the Product(s). The foregoing license does not grant you any distribution rights or other rights to use the API for any other purpose and you agree that you shall not rent, lease, loan, sell, sublicense, assign or otherwise transfer any rights in the API. Polycom retains ownership of the API, and except as expressly set forth herein, no other rights or licenses are granted. Polycom may change, suspend or discontinue providing the API at any time.
4. **Term/Survival.** Without prejudice to any other rights, Polycom may terminate this Agreement if you fail to comply with any of the terms and conditions of this Agreement. In such an event, you must destroy all copies of the API. You may terminate this Agreement at any time by destroying the API. In the event of any termination of this Agreement, Sections 1, 2, 5, and 7-11 shall survive termination.
5. **Development.** Nothing in this Agreement shall impair Polycom’s right to develop, acquire, license, market, promote or distribute products, software or technologies that perform the same or similar functions as, or otherwise compete with any other products, software or technologies that you may develop, produce, market, or distribute. In the absence of a separate written agreement to the contrary, Polycom shall be free to use any information, suggestions or recommendations you provide to Polycom for any purpose, subject to any applicable patents or copyrights.
6. **Harmful Code.** You agree not to include any “Harmful Code” in any products you develop by use of the API, including but not limited to any code that: (i) contains hidden files, “time bombs” or viruses; or (ii) can alter, damage, disclose or erase any data or other computer programs without control of a person operating the computing equipment on which it resides, or (iii) retrieves or collects information without the consent of the user or for any illegal or unauthorized purpose; or (iv) contains a key, node lock, time-out or other function whether implemented by electronic, mechanical or other means which restricts or may restrict use or access to programs or data on the Products, frequency or duration of use, or other limiting criteria; or (v) any code which may restrict, inhibit, disrupt or interfere with the functionality of the Products as provided by Polycom. You agree not to use the API for any illegal or unauthorized purpose.
7. **Marketing/Trademarks.** You are free to market any products you develop using the API, provided you agree not use the Polycom logo, the marks "Polycom," "SoundPoint," "SoundStation," any other marks belonging or licensed to Polycom, or any marks that are confusingly similar to marks belonging or licensed to Polycom in any way except as otherwise expressly authorized by Polycom in each instance. In no event shall you (i) expressly state or imply that any products developed by you were created by or on behalf of Polycom or are being marketed by or on behalf of Polycom; or (ii) expressly state or imply that Polycom has reviewed, sanctioned, or endorsed your product in any way.
8. **No Warranty.** You understand the API provided to you is supplied "AS IS" AND “WITH ALL FAULTS” WITHOUT ANY WARRANTY OF ANY KIND, WHETHER EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, ACCURACY, COMPLETENESS, PERFORMANCE, AND FITNESS FOR A PARTICULAR PURPOSE, AND POLYCOM PROVIDES NO SUPPORT FOR THIS API. You understand that Polycom is under no obligation to provide updates, enhancements, or corrections, or to notify you of any API changes that Polycom may make. In the event you market a product you develop using the API, any obligations, representations or warranties provided by you to an end user shall be solely your obligations, and in no event shall Polycom be responsible to fulfill any such obligations.
9. **Indemnity.** You shall indemnify and hold Polycom harmless from and against any and all costs, damages, losses, liability or expenses (including reasonable attorneys’ fees) arising from your use of the API (including without limitation any actions arising from acts or omissions of your employees or agents) or any failure by you to comply with the terms of this Agreement.
10. **Disclaimer of Liability.** **UNDER NO CIRCUMSTANCES SHALL POLYCOM BE LIABLE FOR SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING WITHOUT LIMITATION, DAMAGES RESULTING FROM DELAY OF DELIVERY OR FROM LOSS OF PROFITS, DATA, BUSINESS OR GOODWILL, ON ANY THEORY OF LIABILITY, WHETHER ARISING UNDER TORT (INCLUDING NEGLIGENCE), CONTRACT OR OTHERWISE, WHETHER OR NOT POLYCOM HAS BEEN ADVISED OR IS AWARE OF THE POSSIBILITY OF SUCH DAMAGES. POLYCOM’S ENTIRE LIABILITY FOR DIRECT DAMAGES UNDER THIS AGREEMENT IS LIMITED TO FIVE DOLLARS (\$5.00).**
11. **Miscellaneous.** If any provision is found to be unenforceable or invalid, that provision shall be limited or eliminated to the minimum extent necessary so that this Agreement shall otherwise remain in full force and effect and enforceable. This Agreement constitutes the entire agreement between the parties with respect to its subject matter and supersedes all prior or contemporaneous understandings regarding such subject matter. No addition to or removal or modification of any of the provisions of this Agreement will be binding upon Polycom unless made in writing and signed by an authorized representative of Polycom.

YOUR USE OF THIS API ACKNOWLEDGES THAT YOU HAVE READ, UNDERSTAND AND AGREE TO BE BOUND BY THE TERMS AND CONDITIONS INDICATED ABOVE.

Polycom, Inc. © 2008. ALL RIGHTS RESERVED.

Corporate Headquarters: Polycom, Inc. 6001 America Center Drive San Jose, CA 95002, USA

www.polycom.com

By downloading the following Sample Applications, you agree to the below end user license agreement.

LICENSE AGREEMENT FOR DEVELOPMENT PURPOSES

This License Agreement for Development Purposes (the "Agreement") is a legal agreement between you and Polycom, Inc., a Delaware corporation ("Polycom").

The software you are about to download (the "Software") comprises sample code that may be useful in the development of applications designed to operate on or in conjunction with Polycom Products.

Polycom is willing to license the Software to you only upon the condition that you accept all of the terms contained in this agreement. Select the "Accept" button at the bottom of the page to confirm your acceptance. If you are not willing to be bound by these terms, select the "Do Not Accept" button and the downloading process will not continue.

PLEASE NOTE:

* POLYCOM OFFERS NO SUPPORT FOR THIS SOFTWARE, AND THE SOFTWARE IS BEING LICENSED WITHOUT DOCUMENTATION, WITHOUT WARRANTY, "AS-IS," AND "WITH ALL FAULTS."

* THE SOFTWARE HAS NOT BEEN TESTED BY POLYCOM AND SHOULD NOT BE LOADED ON PRODUCTION SYSTEMS.

1. GRANT OF LICENSE.

1.1. License. Subject to the terms of this Agreement, Polycom grants to you a nonexclusive, nontransferable license to copy, install, use, and modify the Software, including the Software in source code format, and to produce your own commercial or other purposes derivative works thereof. Except as provided below, this License Agreement does not grant you any rights to patents, copyrights, trade secrets, trademarks, or any other rights related to the Software.

2. DESCRIPTION OF OTHER RIGHTS AND LIMITATIONS.

2.1. Copyright. All title and copyrights in and to the Software and any copies of the Software are owned by Polycom or its suppliers. The Software is protected by copyright laws and international treaty provisions. Title, ownership rights, and intellectual property rights in the Software shall remain in Polycom or its suppliers.

2.2. Ownership of Derivative Works. As between you and Polycom, you will own copyright and other intellectual property rights in derivative works of the Software that you develop.

2.3. Reservation. Polycom reserves all rights in the Software not expressly granted to you in this Agreement.

3. SUPPORT SERVICES.

3.1. No Support Services. Polycom provides no support services for the Software.

4. TERMINATION.

4.1. Termination. Without prejudice to any other rights, Polycom may terminate this Agreement if you fail to comply with any of the terms and conditions of this Agreement. In such event, you must destroy all copies of the Software and all of its component parts. You may terminate this Agreement at any time by destroying the Software and all of its component parts.

5. NO WARRANTY.

THE SOFTWARE IS LICENSED WITHOUT WARRANTY, "AS IS," AND "WITH ALL FAULTS." ALL WARRANTIES, TERMS OR CONDITIONS, EXPRESS OR IMPLIED, EITHER IN FACT OR BY OPERATION OF LAW, STATUTORY OR OTHERWISE, INCLUDING WARRANTIES, TERMS OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, SATISFACTORY QUALITY, CORRESPONDENCE WITH DESCRIPTION, AND NON-INFRINGEMENT, ARE EXPRESSLY DISCLAIMED. POLYCOM NEITHER ASSUMES NOR AUTHORIZES ANY OTHER PERSON TO ASSUME FOR IT ANY OTHER LIABILITY IN CONNECTION WITH THE SALE, INSTALLATION, MAINTENANCE, OR USE OF THIS SOFTWARE.

6. LIMITATION OF LIABILITY.

6.1. Limitations. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL POLYCOM OR ITS SUPPLIERS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR ANY OTHER PECUNIARY

LOSS) ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE, EVEN IF POLYCOM HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. IN ANY CASE, POLYCOM'S ENTIRE LIABILITY SHALL BE LIMITED TO THE GREATER OF THE AMOUNT ACTUALLY PAID BY YOU FOR THE SOFTWARE OR U.S. \$5.00.

7. DISCLAIMER.

7.1. Disclaimer. Some countries, states, or provinces do not allow the exclusion or limitation of implied warranties or the limitation of incidental or consequential damages for certain products supplied to consumers or the limitation of liability for personal injury, so the above limitations and exclusions may be limited in their application to you.

8. EXPORT CONTROLS.

8.1. Export Controls. The Software may not be downloaded or otherwise exported or re-exported (i) into (or to a national or resident of) Cuba, Iraq, Libya, North Korea, Yugoslavia, Iran, Syria, Republic of Serbia, or any other country to which the U.S. has embargoed goods; or (ii) to anyone on the U.S Treasury Department's List of Specially Designated Nationals or the U.S. Commerce Department's Table of Denial Orders. By downloading or using this Software, you are agreeing to the foregoing and you are representing and warranting that you are not located in, under the control of, or a national or resident of any such country or on any such list. If you obtained this Software outside of the United States, you are also agreeing that you will not export or re-export it in violation of the laws of the country in which it was obtained.

9. MISCELLANEOUS.

9.1. Governing Law. This Agreement shall be governed by the laws of the State of California as such laws are applied to agreements entered into and to be performed entirely within California between California residents, and by the laws of the United States. The United Nations Convention on Contracts for the International Sale of Goods (1980) is hereby excluded in its entirety from application to this Agreement.

9.2. Venue for Resolving Disputes. Any disputes relating to this Agreement will be resolved only in the state or federal courts located in Santa Clara County, California. Each of the parties agrees to the exercise over them of the personal jurisdiction of such courts for such purpose.

9.3. U.S. Government Restricted Rights. The Software and documentation are provided with Restricted Rights. The Software programs and documentation are deemed to be "commercial computer software" and "commercial computer software documentation," respectively, pursuant to DFAR Section 227.7202 and FAR 12.212(b), as applicable. Any use, modification, reproduction, release, performance, display, or disclosure of the Software programs and/or documentation by the U.S. Government or any of its agencies shall be governed solely by the terms of this Agreement and shall be prohibited except to the extent expressly permitted by the terms of this Agreement. Any technical data provided that is not covered by the above provisions is deemed to be "technical data commercial items" pursuant to DFAR Section 227.7015(a). Any use, modification, reproduction, release, performance, display, or disclosure of such technical data shall be governed by the terms of DFAR Section 227.7015(b).

9.4. Relationship Between the Parties. The relationship between you and Polycom is that of licensee/licensor. Neither party will represent that it has any authority to assume or create any obligation, express or implied, on behalf of the other party, nor to represent the other party as agent, employee, franchisee, or in any other capacity. Nothing in this agreement shall be construed to limit either party's right to independently develop or distribute software that is functionally similar to the other party's products, so long as proprietary information of the other party is not included in such software.

9.5. Entire Agreement. This Agreement represents the complete agreement concerning this license and may be amended only by a writing executed by both parties. If any provision of this Agreement is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable.

www.polycom.com

Corporate Headquarters: Polycom, Inc. 6001 America Center Drive San Jose, CA 95002 USA