

Spectralink 84-Series Wireless Telephone

Administration Guide

Copyright Notice

© 2012-2016 Spectralink Corporation All rights reserved. Spectralink™, the Spectralink logo and the names and marks associated with Spectralink's products are trademarks and/or service marks of Spectralink Corporation and are 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 Spectralink.

All rights reserved under the International and pan-American Copyright Conventions. No part of this manual, or the software described herein, may be reproduced or transmitted in any form or by any means, or translated into another language or format, in whole or in part, without the express written permission of Spectralink Corporation.

Do not remove (or allow any third party to remove) any product identification, copyright or other notices.

Notice

Spectralink Corporation has prepared this document for use by Spectralink personnel and customers. The drawings and specifications contained herein are the property of Spectralink and shall be neither reproduced in whole or in part without the prior written approval of Spectralink, nor be implied to grant any license to make, use, or sell equipment manufactured in accordance herewith.

Spectralink reserves the right to make changes in specifications and other information contained in this document without prior notice, and the reader should in all cases consult Spectralink to determine whether any such changes have been made.

NO REPRESENTATION OR OTHER AFFIRMATION OF FACT CONTAINED IN THIS DOCUMENT INCLUDING BUT NOT LIMITED TO STATEMENTS REGARDING CAPACITY, RESPONSE-TIME PERFORMANCE, SUITABILITY FOR USE, OR PERFORMANCE OF PRODUCTS DESCRIBED HEREIN SHALL BE DEEMED TO BE A WARRANTY BY SPECTRALINK FOR ANY PURPOSE, OR GIVE RISE TO ANY LIABILITY OF SPECTRALINK WHATSOEVER.

Warranty

The *Product Warranty and Software License and Warranty* and other support documents are available at [+http://support.spectralink.com](http://support.spectralink.com).

Contact Information

US Location

+1 800-775-5330

Spectralink Corporation
2560 55th Street
Boulder, CO 80301
USA

info@spectralink.com

Denmark Location

+45 7560 2850

Spectralink Europe ApS
Bygholm Soepark 21 E Stuen
8700 Horsens
Denmark

infoemea@spectralink.com

UK Location

+44 (0) 20 3769 9800

Spectralink Europe UK
329 Bracknell, Doncastle Road
Bracknell, Berkshire, RG12 8PE
United Kingdom

infoemea@spectralink.com

Contents

About This Guide	11
Who Should Read This Guide?	11
What's New in This Guide	11
Recommended Software Tools	12
Reading the Feature Parameter Tables	12
Product Support	12
Spectralink References	13
<i>Specific Documents</i>	13
Conventions Used In This Document	15
<i>Icons</i>	15
<i>Writing Conventions</i>	16
 Part I: Getting Started	 17
 Chapter 1: Welcome to the Spectralink 84-Series Handsets	 18
Key Features of your Spectralink Handsets	20
 Chapter 2: System Overview	 21
What is SIP?	21
Network Requirements	21
Network Configuration	22
Understanding Spectralink Phone Software Architecture	23
<i>What is the Updater?</i>	23
<i>What is the Spectralink Software?</i>	24
<i>What are the configuration files?</i>	24
<i>What are the resource files?</i>	25
 Part II: Setting Up Your Environment	 26
 Chapter 3: Setting Up Your Device Network	 27
Wireless Device Settings	27
IP Communication Settings	27
Provisioning Server Discovery	28
<i>Supported provisioning protocols</i>	29
Network Configuration Menus	30
<i>Network configuration menu</i>	31
<i>Provisioning server menu</i>	32
<i>Network Interfaces Menu</i>	34

<i>Syslog Menu</i>	39
<i>Login Credentials Menu</i>	40
TLS Security Menu	41
<i>TLS Profile Menu</i>	41
<i>TLS Applications Menu</i>	41

Chapter 4: Setting Up the Provisioning Server..... 43

Why Use a Provisioning Server?	43
<i>Provisioning Server Redundancy</i>	43
Provisioning Server Security Notes	44
Setting up an FTP Server as Your Provisioning Server	44
Downloading Spectralink Software Files to the Provisioning Server	46
<i>Microsoft® Lync® compatibility</i>	46
<i>Spectralink 84-Series Hardware IDs</i>	48
Deploying and Updating Spectralink Handsets with a Provisioning Server	49
<i>Shortcut Method to Deploy Spectralink Handsets with a Provisioning Server</i>	49
<i>Upgrading Spectralink Software</i>	51

Chapter 5: Understanding the Files Written by the Handsets..... 53

Log Files	54
Overrides	54
Contacts	55
Call List	55

Part III: Configuring Features 58

Chapter 6: Features that Cannot be Configured 59

Audio Processing Features	59
<i>Automatic Gain Control</i>	59
<i>Background Noise Suppression</i>	59
<i>Comfort Noise Fill</i>	59
<i>Dynamic Noise Reduction</i>	59
<i>Jitter Buffer and Packet Error Concealment</i>	59
<i>Low-Delay Audio Packet Transmission</i>	60
Call Timer	60
Called Party Identification	60
Connected Party Identification	60
Microphone Mute	60
Synthesized Call Progress Tones	61

Chapter 7: Configurable Features on the User Menus 62

Call Forwarding	62
------------------------------	----

Keypad Lock	66
Multi Key Answer	67
Notification Profiles	67
Time and Date Display.....	77
<i>Synchronizing with SNTP</i>	<i>78</i>
User Preferences Parameters	80

Chapter 8: Features Configured by the Administrator 82

AutoComplete List.....	82
Audio Settings	83
<i>Context Sensitive Volume Control</i>	<i>83</i>
<i><voice.volume/>.....</i>	<i>83</i>
<i><voice/></i>	<i>84</i>
<i><rxQoS/>.....</i>	<i>84</i>
Automatic Off-Hook Call Placement.....	85
Background Images.....	86
<i>Configuring Background Images</i>	<i>86</i>
Feature and Basic Settings Menu Password	87
Call Hold	88
Call Handling Features	91
<i>Call Park and Retrieve</i>	<i>92</i>
<i>Call Waiting Alerts.....</i>	<i>92</i>
<i>Calling Party Identification.....</i>	<i>92</i>
<i>Missed Call Notification.....</i>	<i>93</i>
<i>Call Transfer</i>	<i>93</i>
<i>Call Lists</i>	<i>95</i>
<i>Miscellaneous Call Handling Parameters</i>	<i>95</i>
CMS 2.0.....	97
Conference Calls	97
Corporate Directory	98
Default Ring Tones and Alert Tones.....	101
<i>Call Progress Patterns</i>	<i>104</i>
Do Not Disturb	106
Dual Tone Multi-Frequency (DTMF) Tones.....	106
<i>DTMF Event RTP Payload</i>	<i>107</i>
Emergency Calls.....	108
<i>Emergency Dial via Authorized Call menu.....</i>	<i>108</i>
<i>Emergency Dial via Duress Button.....</i>	<i>109</i>
Enhanced Feature Keys	111
<i>Guidelines for Configuring Enhanced Feature Keys</i>	<i>112</i>
<i>Understanding Macro Definitions</i>	<i>115</i>
<i>Macro Action.....</i>	<i>115</i>
<i>Prompt Macro Substitution</i>	<i>117</i>
<i>Expanded Macros</i>	<i>118</i>

<i>Special Characters</i>	118
Features Softkey Menu Options Customization	120
<i>Example Softkey Configurations</i>	122
Handsfree Settings	125
<i>Bluetooth Headset Support</i>	126
Language Support	126
Local Contact Directory	129
<i>Provisioning the Seed Directory</i>	130
<i>Configuring the Contact Directory</i>	132
<i>Editing the Users' MACaddress-directory.xml File</i>	133
<i>Specialized Caller Treatments</i>	133
Location Services (Ekahau)	135
Microsoft Exchange Calendar Integration	136
Open Application Interface	138
Passwords – User and Administrator	139
Personal Alarms	141
<i>Administrator Configurable Options</i>	142
<i>User Experience</i>	144
<i>Integration with Third Party Applications</i>	147
<i>XML API Detail</i>	151
<i>Viewing an Alarm Event</i>	151
<i>Configuration Template</i>	152
Phone Lock	153
Provisional Polling of Spectralink Handsets	155
Push-to-talk and Group Paging	156
<i>Push-to-talk</i>	157
<i>Group Paging</i>	159
Quick Barcode Connector Application	161
Registrations	161
<i>Multiple Registrations</i>	164
<i>Multiple Concurrent Calls</i>	165
<i>Flexible Call Appearances</i>	167
User Profiles	167
<i>Placing Authorized (Emergency) Calls without Logging In</i>	169
Voicemail Integration	170
<volpProt/>	172
Web Browser	178
<mb/>.....	181
<oai/>.....	182

Chapter 9: Web Application Parameters..... 183

Application menu configuration <apps.>.....	183
Web browser parameters <mb.>.....	184
State Polling Parameters <apps.statePolling.>	185

Push Request Parameters <apps.push.>	185
Telephony Notification Parameters (apps.telNotification.>	187
Open Application Interface parameters <oai.>	189
Sample Configurations	189
<i>Push</i>	189
<i>Telephony Notifications</i>	190
<i>State Polling</i>	190
<i>Personal Alarms</i>	191

Chapter 10: System-Level Parameters 192

Configuration File Encryption	192
Understanding Digital Certificates	193
<i>About Digital Certificates</i>	195
<i>Types of certificates</i>	195
<i>Configuring certificates</i>	196
<i>Generating a Certificate Signing Request</i>	207
<i>Downloading Certificates to a Spectralink Phone</i>	208
DNS SIP Server Name Resolution	208
<i>Behavior When the Primary Server Connection Fails</i>	209
Incoming Signaling Validation	211
Instant Messaging	212
IP Type-of-Service	215
<qos/>	215
Logging Parameters	216
<level/> <change/>and<render/>	217
<sched/>	218
Microsoft Lync Server 2013/2010 Integration	219
Network Address Translation (NAT)	219
Provisioning Server System Settings	220
<request/>	221
Security <sec/>	221
<sctp/>	221
<dot1x><eapollogoff/>	222
Secure Real-Time Transport Protocol	222
Server Redundancy	225
<i>Terminology</i>	225
<i>About the Optional Failover Behaviors</i>	226
<i>Fallback Deployments</i>	229
<i>Failover Deployments</i>	230
<i>DNS Server Unavailability</i>	230
<i>Redundancy Parameters</i>	230
Supporting 802.1X Authentication	233
<tcplpApp/>	235
<dhcp/>	235

<code><dns/></code>	235
<code><ice/></code>	235
<code><keepalive/></code>	236
Tones <code><tones/></code>	237
<code><chord/></code>	237
Web Configuration Utility	238
<code><httpd/></code>	238

Chapter 11: Special Use Cases 239

Acoustic Echo Cancellation	239
Audio Codecs	239
Band Steering	240
Bridged Line Appearance	242
Local Digit Map	243
<i>Understanding Digit Map Rules</i>	244
Location Values for E.911 Services	249
Real-Time Transport Protocol Ports	250
Shared Line Appearances	251
<i>Shared Call Appearance Signaling</i>	252
Static DNS Cache	253
<i>Using Static DNS Cache for Redundancy</i>	258
DNS Cache <code><dns/></code>	259
NAPTR <code><NAPTR/></code>	259
SRV <code><SRV/></code>	260
A <code><A/></code>	260
Voice Activity Detection	264

Part IV: Troubleshooting and Maintaining your Deployment 265

Chapter 12: Troubleshooting Your Spectralink Handsets 266

Troubleshooting Flow Diagram	267
Understanding Error Message Types	268
<i>Updater Error Messages</i>	268
<i>Spectralink Software Error Messages</i>	270
Status Menu	275
Log Files	276
<i>Logging Modules</i>	278
<i>Major categories of WLAN entries</i>	278
Managing the Phone's Memory Resources	280
<i>Identifying Symptoms</i>	280
<i>Checking the Phone's Available Memory</i>	281

<i>Managing the Phone Features</i>	282
Testing Phone Hardware	283
Uploading a Phone's Configuration	284
Network Diagnostics	284
Network Protocols and Ports Used on Spectralink Handsets	285
Power and Startup Issues	286
Key Pad Issues	286
Screen and System Access Issues	287
Calling Issues	287
Display Issues	288
Audio Issues	288
Upgrading Issues	289

Chapter 13: Miscellaneous Maintenance Tasks..... 291

Encrypting Configuration Files	291
<i>Comparing encrypted and unencrypted files</i>	295
<i>Decrypting existing configuration files</i>	295
<i>Changing an existing key</i>	296
<i>Log messages</i>	296
Multiple Key Combinations	297
<i>Rebooting the Phone</i>	297
<i>Resetting to factory defaults</i>	298
<i>Updating log files</i>	298
<i>Setting base profile</i>	298
Default Feature Key Layouts	299
Parsing Vendor ID Information	300
Product Model Number and Hardware ID Mapping	301
Capturing the Phone's Current Screen	302

Part V: Appendices..... 303

Appendix A: Ringtone Pattern Names and Sound Effects Parameters 304

Ringer Patterns	304
Ring Tones <rt/>	305

Appendix B: Session Initiation Protocol (SIP) Information 307

RFC and Internet Draft Support	307
<i>Request Support</i>	309
<i>Header Support</i>	309
<i>Response Support</i>	312
<i>Hold Implementation</i>	314

<i>Reliability of Provisional Responses</i>	315
<i>Transfer</i>	315
<i>Third party call control</i>	315
<i>SIP for Instant Messaging and Presence</i>	315

Appendix C: Open Source Information 316

OFFER for Source for GPL and LGPL Software	316
<i>Contact Information for Requesting Source Code</i>	316

Appendix D: Library of <device/> Settings..... 317

Appendix E: Trusted Certificate Authority List..... 324

Appendix F: Spectralink Certificates 335

About This Guide

This Spectralink 84-Series Administration Guide provides advanced instructions for installing, provisioning, and administering Spectralink handsets. It is a companion to the *Spectralink 84-Series Deployment Guide* which is your essential reference for understanding how to provision and deploy Spectralink 84-Series handsets in any environment. This guide expands upon the information provided in the Deployment Guide and provides additional data about how the software works and provides descriptions of all applicable parameters. Specifically, this Administration Guide will help you perform the following tasks:

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

Who Should Read This Guide?

System administrators and network engineers should read this guide for advanced information on configuring and understanding Spectralink 84-Series handsets. This guide describes administration-level tasks and is not intended for end users.

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

- The information in the *Spectralink 84-Series Deployment Guide* is not duplicated in this document. This document expands upon the basic configuration settings in the Deployment Guide and this document assumes you are familiar with Deployment Guide information.
- Computer networking and driver administration for your operating system
- An XML editor
- The XML-based configuration file format that the Spectralink Software and its supported handsets use.

What's New in This Guide

The content in this guide has been significantly revised from the Polycom UCS version for use with the Spectralink 84-Series handsets. It is designed for clarity and to provide more information to system administrators who are already familiar with deploying Spectralink 84-Series handsets.

Recommended Software Tools

Spectralink 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.

See the *Spectralink 84-Series Deployment Guide* for a discussion on XML editor options, usefulness and limitations.

Reading the Feature Parameter Tables

Each of the feature descriptions discussed in *Part III: Configuring Features* includes a table of parameters that you configure to make the features work. Although there are three provisioning methods you can use to configure a feature: a centralized provisioning server, the Web Configuration Utility, or the local handset user interface, this document emphasizes the central provisioning server method. It is the preferred method for deploying advanced configurations such as those covered in this document as it is the only method that is available for every feature. The Web Configuration Utility and the local handset user interface do not provide access to all features.

The central provisioning server method requires you to configure parameters located in template configuration files that Spectralink provides in XML format.

We recommend using the search feature of your XML editor to locate the parameters you need to find.

Product Support

Spectralink wants you to have a successful installation. If you have questions please contact the Customer Support Hotline at 1-800-775-5330.

The hotline is open Monday through Friday, 6 a.m. to 6 p.m. Mountain time.

For Technical Support: <mailto:technicalsupport@spectralink.com>

For Knowledge Base: <http://support.spectralink.com>

For Return Material Authorization: <mailto:nalarma@spectralink.com>

Spectralink References

All Spectralink documents are available at <http://support.spectralink.com>.

The screenshot shows the Spectralink Support website. At the top, there is a navigation bar with links for Partner Access, Spectralink.com, Contact Support, and a Search icon. Below this is the Spectralink logo with the tagline 'solving every day' and the word 'support'. To the right of the logo are links for PRODUCT RESOURCES, RMAs, SERVICE REQUESTS, and CUSTOMER MANAGEMENT. The main heading is 'Welcome to Spectralink Support' with a subtext 'Find resources for your product, or log in for more support options.' Below this is a 'PRODUCT RESOURCES' section with a search bar for product documents and downloads. The search bar has dropdowns for 'Product Category' (set to 'Wi-Fi') and 'Product Type' (set to '- Any -'), and a 'FIND' button. To the right of the search bar is a list of links: 'All Documents & Downloads', 'Feature Requests', 'Product Alerts', 'Service Policies', 'FAQs', and 'Contact Support'. Below the search bar are two sections: 'RMAs AND SERVICE REQUESTS' and 'CUSTOMER MANAGEMENT', each with a lock icon. The 'RMAs AND SERVICE REQUESTS' section contains links for 'RMA Status', 'My Service Requests', 'RMA Forms', 'My Company's Service Requests', 'RMA Requests', 'Repair Pricing', and 'My Company's RMAs'. The 'CUSTOMER MANAGEMENT' section contains links for 'Warranty and Entitlement Lookup', 'My Company's Entitlements', and 'Batch Warranty and Entitlement Lookup'. At the bottom, there is a copyright notice: '© 2013 Spectralink Corporation, All rights reserved. Terms and Conditions | Product Warranty'.

To go to a specific product page:

Select the Product Category and Product Type from the dropdown lists and then select the product from the next page. All resources for that particular product are displayed by default under the All tab. Documents, downloads and other resources are sorted by the date they were created so the most recently created resource is at the top of the list. You can further sort the list by the tabs across the top of the list to find exactly what you are looking for. Click the title to open the link.

Specific Documents

Spectralink 84-Series Wireless Telephone Deployment Guide This document introduces deployment concepts and the methods of provisioning the 84-Series handsets in any type of facility. It is the fundamental text and a prerequisite to this Administration Guide, especially for administrators who are new to the Spectralink 84-Series handsets or who may wish a refresher course.

Spectralink Deploying Enterprise-Grade Wi-Fi Telephony This document covers the security, coverage, capacity and QoS considerations necessary for ensuring excellent voice quality within enterprise Wi-Fi networks.

Best Practices Guide to Network Design Considerations for Spectralink Wireless Telephones This document provides recommendations for ensuring that a network environment is adequately optimized for use with Spectralink Wireless Telephones. It provides detailed information on wireless LAN layout, network infrastructure, QoS, security and subnets and identifies issues and solutions based on Spectralink's extensive experience in enterprise-class Wi-Fi telephony. This document has a brief discussion about wireless security.

Understanding Wireless Security on Your Spectralink 84-Series Wireless Phones Provides more information and assistance in determining which security method to use.

Barcode Administration Guide Provides information about barcode symbologies and how to configure and implement the barcode feature on the handset. The *Spectralink 84-Series User Guide* also contains information about using the barcode feature.

Quick Barcode Connector Administration Guide Provides instruction for implementation of the barcode application. The *Spectralink 84-Series User Guide* also contains information about deploying the barcode feature.

The *Spectralink 84-Series User Guide* offers comprehensive instructions on using each of the features deployed on the handsets.

For information on IP PBX and softswitch vendors, see the *Spectralink 84-Series Call Server Interoperability Guide*.

For information about combining Polycom desksets and Spectralink 84-Series handsets in the same facility, see the *Interoperability Guide: Spectralink 84-Series Wireless Telephones and Polycom Desksets*.

AP Configuration Guides explain how to correctly configure access points and WLAN controllers (if applicable) and identify the optimal settings that support Spectralink 84-Series handsets.

Technical Bulletins and Feature Descriptions explain workarounds to existing issues and provides expanded descriptions and examples.

Release Notes describe the new and changed features, and resolved issues in the latest version of the software. Find them in the Downloads section of the support site.

Spectralink 84-Series Wireless Telephones Web Developer's Guide assists with the development of applications that run on the browser on the Spectralink 84-Series Wireless Handsets.

Spectralink 8000 Open Applications Interface (OAI) Gateway Administration Guide provides information about deploying third party applications through the OAI gateway interface.

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

Conventions Used In This Document

Icons

Icons indicate extra information about nearby text.



Warning

The *Warning* icon highlights an action you must perform (or avoid) to avoid exposing yourself or others to hazardous conditions.



Caution

The *Caution* icon highlights information you need to know to avoid a hazard that could potentially impact device performance, application functionality, successful feature configuration and/or affect handset or network performance.



Note

The *Note* icon highlights information of interest or important information that will help you be successful in accomplishing a procedure or understanding a concept.



Tip

The *Tip* icon highlights information that may be valuable or helpful for users to know, such as special techniques, shortcut methods, or information that will make user tasks easier to perform.



Web

The *Web Info* icon highlights supplementary information available online such as documents or downloads on support.spectralink.com or other locations.



Timesaver

A time-saving tip is typically used to mention or highlight a faster or alternative method for users who may already be familiar with the operation or method being discussed.



Admin Tip

This tip advises the administrator of a smarter, more productive or alternative method of performing an administrator-level task or procedure.



Power User

A Power User Tip is typically reserved for information directed specifically at high-level users who are familiar with the information or procedure being discussed and are looking for better or more efficient ways of performing the task. For example, this might highlight customization of a feature for a specific purpose.



Troubleshooting

This element can be used in any type of document and is typically used to highlight information to help you solve a relevant problem you may encounter, or to point to other relevant troubleshooting reference information.



Settings

The Settings icon highlights information to help you zero in on settings you need to choose for a specific behavior, to enable a specific feature, or access customization options.

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 menu navigation. 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 <code>parameter.*</code> , the text is referring to all parameters beginning with <code>parameter</code> .

Part I: Getting Started

Part I gives you an overview of the Spectralink 84-Series handsets and of the Spectralink Software.

Chapter 1: Welcome to the Spectralink 84-Series Handsets

This chapter introduces Spectralink 84-Series handsets used with Spectralink Software version 4.2.0 and above.

The Spectralink family of handsets provides a powerful, yet flexible wireless IP communications solution for Ethernet TCP/IP networks. Not only do the handsets 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 handsets can also support advanced functionality, including multiple call and flexible line appearances, HTTPS secure provisioning, presence, custom ringtones, and local conferencing.



Note: Indoor use only

This device is intended for indoor use only.



Caution: Product compatibility/safety

Spectralink 84-Series handsets are intended for operation only with Spectralink 84-Series battery packs and Spectralink 84-Series chargers. These Spectralink components are critical to product safety certification and may not be substituted. Representative samples of the Spectralink 84-Series handsets, battery packs and chargers have been tested as a complete system by an independent testing organization and have been certified by that organization to meet applicable safety standards. Use or operation of the Spectralink 84-Series handsets with batteries or chargers other than those authorized by Spectralink has not been tested or safety certified. Spectralink 84-Series handsets, battery packs or chargers used or operated with products not authorized by Spectralink are not covered by the Spectralink Limited Product Warranty.



Caution: Use authorized components only

Only Spectralink 84-Series battery packs and Spectralink 84-Series chargers are authorized for use or operation with Spectralink 84-Series handsets.

Only Spectralink 84-Series battery packs are authorized for use or operation with a Spectralink 84-Series charger and are not authorized to be used or operated in any other charger.

Spectralink 84-Series Wireless Telephones were originally developed in conjunction with Polycom Inc. In September 2012 the Spectralink 84-Series Wireless Telephone software code and Polycom UCS deskset software code were split into two separate streams. The same code no longer serves both Spectralink Wireless Telephones and Polycom wired desksets.



Using Spectralink Handsets and Polycom Desksets in a combined environment

Special configuration steps need to be taken in environments where both Spectralink and Polycom phones are deployed. For more information on using Spectralink 84-Series handsets and Polycom wired desksets in a facility see the *Interoperability Guide: Spectralink 84-Series Wireless Telephones and Polycom Desksets*.

From an administrator's perspective, the handsets 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.

If you want to begin setting up your Spectralink handsets on the network, go to [Setting Up Your Device Network](#).

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



Support for Spectralink Handsets

You can find all documentation for all Spectralink handsets on [Spectralink Support Website](#). For more information, contact your Spectralink distributor.

Spectralink 8440/8441



Spectralink 8450/8452/8453



Key Features of your Spectralink Handsets

Spectralink handsets running Spectralink Software include the following key features:

- Award winning sound quality with a full-duplex speakerphone
 - Permits natural, high-quality, two-way conversations
 - Supports HDVoice
- Easy-to-use
 - An easy transition from traditional PBX systems into the world of IP Communications
 - Four context-sensitive softkeys for further menu-driven activities
- Platform independent
 - Supports multiple protocols and platforms enabling standardization of one handset for multiple locations, systems, and vendors
- Faster Boot Time
 - The time between handset reboot and obtaining a dial tone has been noticeably reduced.
- Field upgradeable
 - Upgrade handsets as standards develop and protocols evolve
 - Extends the life of the handset 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
 - 240 x 320 pixel graphical color LCD
- Multiple language support
 - 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.
- Web Browser
 - Supports a subset of XHTML constructs that run like any other Web browser
- XML status/control API
 - Ability to poll handsets for call status and device information
 - Ability to receive telephony notification events

Chapter 2: System Overview

This chapter provides an overview of the Spectralink Software, providing an understanding of how the handsets are deployed within the greater LAN and wireless LAN configuration. To begin setting up your Spectralink handsets, refer to the *Spectralink 84-Series Deployment Guide* and review the Infrastructure chapter.

The Spectralink handsets are deployed in an 802.1X wireless environment.



Deploying Spectralink Handsets in a Completely Wireless Environment

For more information on using these handsets in a completely wireless environment, see the *Spectralink 84-Series Wireless Telephone Deployment Guide*.

What is SIP?

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.

Network Requirements

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

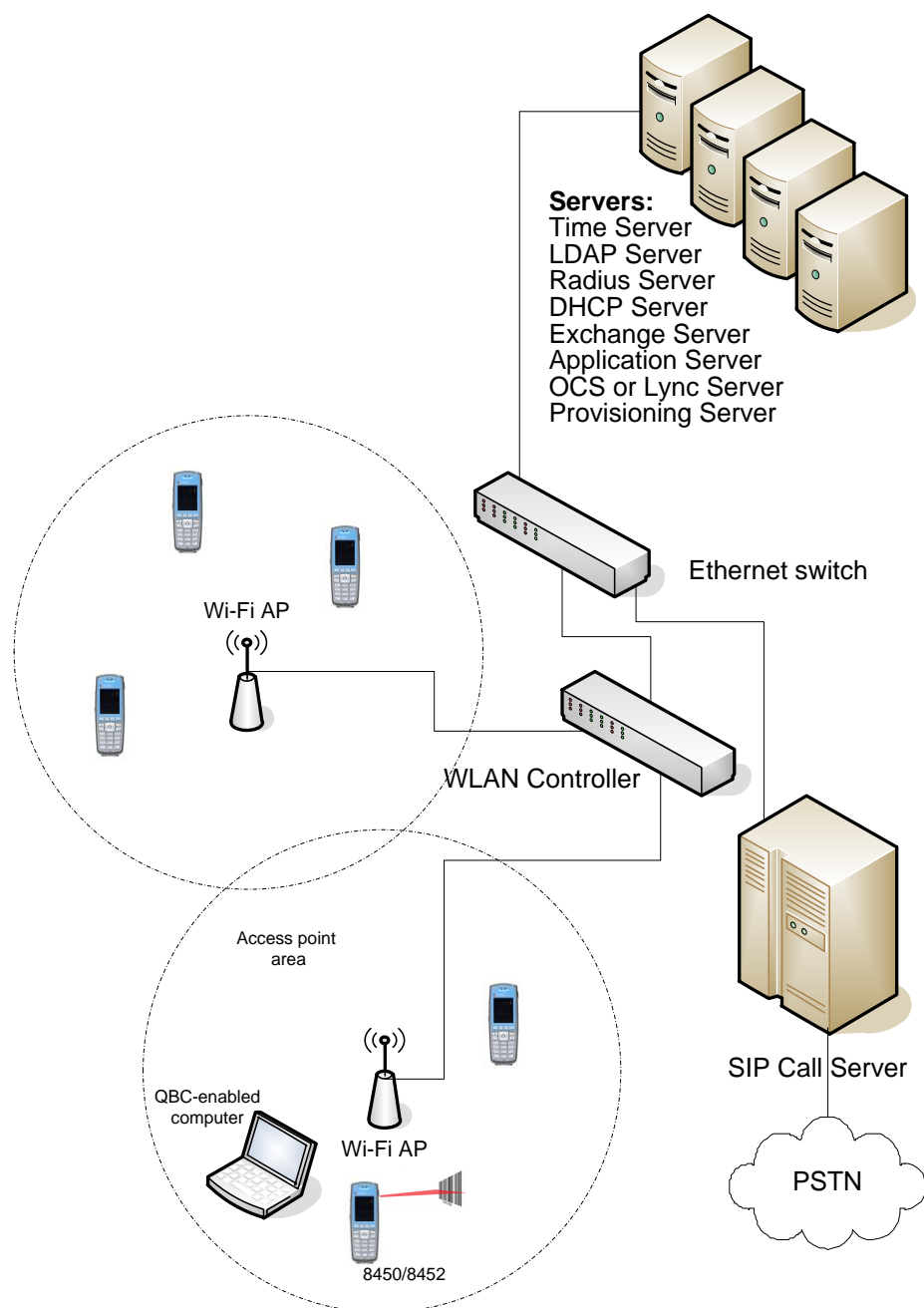
- A working IP network
- Routers configured for VoIP
- VoIP gateways configured for SIP
- The latest (or a compatible version) Spectralink Software image
- An active, configured call server to receive and send SIP messages

For information on IP PBX and softswitch vendors, see the *Spectralink 84-Series Call Server Interoperability Guide*.

Network Configuration

Many desktop 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. Spectralink wireless handsets, however, connect to a WLAN to send and receive all data using 802.11a/b/g/n technology to access telephony resources on the wired LAN.

There are many ways to set up a phone network using Spectralink wireless handsets and the diagram shown next is just one example of a network setup.



Understanding Spectralink Phone Software Architecture

The Spectralink handset software is made of four basic components:

- **Updater** The software that loads & runs first when the handset is powered on
- **Spectralink Software** The software that implements the handset functions and features
- **Configuration files** The files that contain the handset's parameter settings
- **Resource files** Optional files that contain settings for advanced features

Figure 2-2: Spectralink phone software



What is the Updater?

The Updater is a small application that resides in the flash memory on the handset. The Updater is installed at the factory and is already installed on your new Spectralink handsets.

When you start/boot/reboot the handset, 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 handset 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 Spectralink 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 Spectralink Software will take over.

What is the Spectralink Software?

The Spectralink Software manages the protocol stack, the digital signal processor (DSP), the user interface, and the network interaction. The Spectralink Software implements the following functions and features on the handsets:

- VoIP signaling for a wide range of voice telephony functions using SIP signaling for call setup and control
- Industry standard security techniques for ensuring that all provisioning, signaling, and media transactions are robustly authenticated and encrypted across the WLAN
- Advanced audio signal processing for handset, headset, and speakerphone communications using a wide range of audio codecs
- Flexible provisioning methods to support single handset, 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.

Each release of software includes a new image file.

Both the Updater and Spectralink Software run on all 84-Series handset models that Spectralink currently supports.

What are the configuration files?

The Spectralink Software that you download contains configuration file templates, valid XML files that you can edit using an XML editor. These files contain all the parameters explained in this document that provision the handsets with features and settings. The configuration files are very flexible: you can rearrange the parameters within the file, move parameters to new files, or create your own configuration files with only those parameters you want. This flexibility is useful when you want to apply the same features and settings to a large number of handsets. Use of the configuration files to provision the handsets 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 handsets to read the same set of files. You can also configure a subset of handsets to use only specific files, thereby deploying different handsets with different sets of features.

Spectralink recommends that you configure handsets using the centralized provisioning method. However, you can also configure individual handsets using the handset's menu system, accessible through the local user keypad interface, or you can configure select parameters by using the Web Configuration Utility.

You will need to keep in mind that there is a hierarchy among the configuration methods and settings. 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 handsets. Spectralink strongly recommends becoming familiar with each of the configuration methods.

Override files are maintained on the central provisioning server. See [Understanding the Files Written by the Handsets](#) for additional information on how precedence is used by the handsets.

Configuration Methods

You can make changes to the handset'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 handset 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 handset user interface

Web Configuration Utility

Central Provisioning Server

Default values (if Null then the value will be obtained from a higher priority method.)

Each of these configuration methods is detailed in [Part III: Configuring Features](#).

What are the resource files?

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

Examples of resource files include:

- Language dictionaries
- Ringtones
- Contact directories
- Custom backgrounds

If you need to remove resource files from a handset at a later date - for example, if you are giving the handset to a new user - you will have to apply factory default settings to that handset. For instructions on how to reset your handset to factory default settings, see the *Spectralink 84-Series Deployment Guide*.

Part II: Setting Up Your Environment

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

Part II consists of the following chapters:

- Chapter 3: Setting Up Your Device Network
- Chapter 4: Setting Up the Provisioning Server
- Chapter 5: Understanding the Files Written by the Handsets

Chapter 3: Setting Up Your Device Network

The Spectralink 84-Series Wireless Handsets operate on a Wi-Fi LAN (WLAN). Local area network design varies by organization and Spectralink handsets can be configured to accommodate a number of network designs. This chapter shows you several automated and manual ways to configure Spectralink handsets to operate in a LAN.

See the *Spectralink 84-Series Deployment Guide* for detailed information about how the handset authenticates and associates with the WLAN.

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

Wireless Device Settings

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). See the *Spectralink 84-Series Wireless Telephone Deployment Guide* for full information about using the USB interface and basic wireless settings.

More advanced wireless settings that may need to be set up to connect your device to the Wireless LAN (WLAN) are located in the [Wi-Fi Menu](#) section.

IP Communication Settings

When the handset 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: Novice administrator?

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 handset UI, or to pre-provision using a `device.set` capability.

When making the DHCP request the handset will include information in Option 60 that can assist the DHCP server in delivering the appropriate settings to the device.



Timesaver: Reducing repetitive data entry

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

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

Option	SIP Parameter	Meaning
1	NA	Subnet mask
3	NA	Default gateway
6	DNSSVR	DNS server
7	LOGSRVR	Syslog server logging
15	DOMAIN	Domain name
42	SNTPSRVR	NTP Server
43	sec.TLS.customCaCert.x	Auto discovery of the root CA certificate. If this setting is unavailable, set the parameter per this guide.
66	TFTPSVR	TFTP server



Web Info: RFC information on DHCP options

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



Settings: Overriding the SNTP values set by DHCP

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](#).

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 handset 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](#).

Provisioning Server Discovery

After the handset 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 Administrator?

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 handset supports several methods to 'discover' this provisioning server:

- **Static** You can manually configure the server address from the handset's user interface or the Web Configuration Utility, or you can pre-provision the handset with an initial provisioning server. The parameters are:
`device.prov.serverName.set="1"` and `device.prov.serverName=""` in a configuration file.
- **DHCP** DHCP option 66 is used to provide the address or URL of the provisioning server.
- **DHCP INFORM** The handset 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](#).

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

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, handsets 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.

Supported protocols include: ftp, ftps, tftp, http and https.

In some cases a config file might need to be secured. You can use unique credentials to connect to a server and include the transfer protocol in the configuration file name. 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.



Settings: Choosing a valid URL

A URL should contain forward slashes (not 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. Spectralink Software is not compatible with active FTP.

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



Web Info: Viewing trusted certificate authorities

For more information, see [Appendix E: Trusted Certificate Authority List](#) and Technical Bulletin CS-13-06: *Using custom certificates with Spectralink 8400 handsets*.

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 handset 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.mspx?mfr=true>.

Network Configuration Menus

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 handset boots up, press the **Cancel** softkey, and press the **Setup** softkey to launch the setup menu. To access the setup menu, you will have to enter the administrator's password.
- **After your handset starts and is running Spectralink Software.** The network configuration menu is accessible from the handset's main menu. Select **Settings> Advanced Settings> [enter password]> Administration Settings> Network Configuration**. To access the **Advanced Settings** menu, you will have to enter the administrator's password which is 456 by default.



Admin: Changing the default administrator password

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

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

- Main Menu
- Provisioning Server Menu
- Network Interfaces Menu (Ethernet Menu)
- TLS Menu
- Syslog Menu

Use the softkeys, 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.



Settings: Resetting network configurations

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

Network configuration menu

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

Name	Possible Values
Provisioning server menu	
See Provisioning Server Menu .	
Network interfaces menu or Ethernet menu	
See Network Interfaces Menu (Ethernet Menu) .	
SNTP address	Dotted-decimal IP address OR Domain name string
The Simple Network Time Protocol (SNTP) server the handset 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.	

Name	Possible Values
DNS server	Dotted-decimal IP address
The primary server the handset directs Domain Name System (DNS) queries to.	
DNS INFORM server	Dotted-decimal IP address
The secondary server to which the handset directs DNS queries.	
DNS domain	Domain name string
The handset's DNS domain.	
Hostname	hostname
The DHCP client hostname.	
Syslog menu	
See Syslog Menu .	
Base profile	Generic, Lync

Provisioning server menu

The following configuration parameters can be modified on the Provisioning Server Menu. **Settings> Advanced Settings> [enter password]> Administration Settings> Network Configuration> Provisioning Server:**

Name	Possible Values
DHCP menu	
See DHCP Menu . Note: This menu is disabled when the DHCP client is disabled.	
Server type	0=FTP, 1=TFTP, 2=HTTP, 3=HTTPS, 4=FTPS
The protocol that the handset will use to obtain configuration and handset application files from the provisioning server. See Supported Provisioning Protocols .	
Note: Active FTP is not supported for BootROM version 3.0 or later. Passive FTP is supported. Only implicit FTPS is supported.	
Server address	Dotted-decimal IP address OR URL
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 handset 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 handset 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 <i>Spectralink 84-Series Wireless Telephone Deployment Guide</i> .	
Note: ":", "@", or "/" can be used in the user name or password if they are correctly escaped using the method specified in RFC 1738.	
Server user	String
The user name requested when the handset logs into the server (if required) for the selected Server type .	
Note: If the <i>Server address</i> is a URL with a user name, this will be ignored.	

Name	Possible Values
Server password	String
The default password is 456.	
File Transmit Tries	1 to 10 Default 3
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.)	
Retry Wait	0 to 300 seconds Default 1
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.	
Tag SN to UA	Disabled, Enabled
If enabled, the handset's serial number (MAC address) is included in the User-Agent header of HTTP/HTTPS transfers and communications to the browser. The default value is Disabled.	
Upgrade Server	Non-editable string (auto-populated by the Web Configuration Utility)
The Upgrade server is an alternate way of getting software updates into the handset through the Web Configuration Utility. It is a completely different process than using a provisioning server method. When a value is displayed in this field, it is the address/URL that has been accessed for software updates through the Web Configuration Utility. This value is also stored in the handset's override file on the provisioning server. If this field in the handset menu is populated then you cannot get code onto the phone from any other method than using the WebUI upgrade method because handset settings have highest precedence and this setting is basically a mirror of the override file. If you want to download code into a particular handset using a provisioning server, clear the value set by the WebUI, or edit the override file parameter to "" which will also delete the setting in the handset.	



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:

Name	Possible Values
Boot Server	0=Option 66, 1=Custom, 2=Static, 3=Custom+Option 66
<p>Option 66: The handset 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 handset 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 handset 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 handset 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>	

Name	Possible Values
	<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 handset 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.
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 handset 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 handset 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 Using DHCP Vendor Identifying Options with Spectralink Handsets.</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>	



Multiple DHCP INFORM Servers

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

Network Interfaces Menu

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

- [Wi-Fi Menu](#)
- [USBNet Menu](#)

Wi-Fi Menu

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

Table 3-10: Wi-Fi Menu

Name	Possible Values
Enabled	Yes, No
A flag to determine if the wireless interface is enabled or not.	

<i>Name</i>	<i>Possible Values</i>
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 handset's IP address.	
Note: This option is not available when the DHCP parameter is Enabled.	
Subnet Mask	Dotted-decimal subnet mask
The handset's subnet mask.	
Note: This option is not available when the DHCP parameter is Enabled.	
IP Gateway	Dotted-decimal IP address
The handset's default router.	
AC Required	Yes, No
A flag to determine if handsets will connect only to APs (access points) that enforce access control (Wi-Fi Multimedia Admission Control [WMM-AC]). (See Caution note below.)	
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 .	
Radio	
See Radio Menu .	



Caution: WMM-AC not supported by 87-Series handsets

When deploying both 84-Series and 87-Series handsets in the same facility using the same Wireless LAN, Wi-Fi Multimedia Admission Control (aka access control, AC or WMM-AC) must be disabled in any handset parameters and APs as it is not supported by 87-Series handsets. Any parameter that requires or enforces AC must be disabled.

WEP Menu

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

Table 3-11: WEP Menu

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

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 3-22: WPA (2) PSK Menu

Name	Possible Values
PSK Type The pre-shared key type.	0=Passphrase (default), 1=Hexadecimal key
Passphrase The authentication passphrase. Note: This parameter is unavailable when PSK Type is 1.	8 to 63 character ASCII string
Key The authentication key. Note: This parameter is unavailable when PSK Type is 0.	256 bit hexadecimal string

WPA2-Enterprise Menu

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

Table 3-3: WPA2-Enterprise Menu

Name	Possible Values
Fast Roaming Method	0=Opportunistic Key Caching (OKC) , 1= Cisco Centralized Key Management (CKM)
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	1=EAP-TLS, 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.	

Table 3-8: 802.1X Menu

Name	Possible Values
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
The selected EAP type to be used for authentication. For more information, see Supporting 802.1X Authentication .	
User ID	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.	
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 handset.

Radio Menu

You can modify the following parameters from the Radio menu:

Table 3-44: Radio Menu

<i>Name</i>	<i>Possible Values</i>
Regulatory Domain	0, 1, 2 or 10
Available values specify the regulatory domain. The supported values are 1 (North America), 2 (Europe) and 10 (Australia). If 0, 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 3-55: 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. The “7” setting is also called Auto in some menus and is the maximum allowable power for that channel and data rate. If no maximum is set, the handset uses the P5 settings for each channel activated. For more information, see the device.wifi.radio.band5GHz.subBandx.txPower set of parameters in Appendix D.	

2.4 GHz Menu

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

Table 3-6: 2.4 GHz Menu

Name	Possible Values
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 sub-band. The “7” setting is also called Auto in some menus. If no maximum is set, the handset uses the P5 settings for each channel activated. Note that ESTI regulations limit the maximum setting in Europe. If P4 or above is selected for domain 2, the handset will broadcast at the maximum allowable power which is 12 mW.. For more information, see the <code>device.wifi.radio.band2.4GHz.subBandx.txPower</code> set of parameters in Appendix D.	

USBNet Menu

You can modify the following parameters from the USBNet menu:

Table 3-77: USBNet Menu

Name	Possible Values
Enabled	1=Yes, 2=No
A flag to determine if USB networking is supported. USBnet is used by SLIC and USB Setup for initial provisioning. SLIC disables it when the files are produced. Parameter: <code>device.usbnet.enabled</code>	
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 handset's subnet mask. For Spectralink handsets, the default value is 255.255.0.0 .	
IP Gateway	Dotted-decimal IP address
The handset'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.	

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 3-83: Syslog Menu

Name	Possible Values
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 handset 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>device.syslog.renderLevel</code> and can be a lower value. See Appendix D for device parameters. Note: Use left and right arrow keys to change values when using the Admin menu on the handset.	
Prepend MAC Address	Enabled, Disabled
If enabled, the handset's MAC address is prepended to the log message sent to the syslog server. Spectralink recommends enabling this parameter.	

Login Credentials Menu

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

Table 3-9: Login Credentials Menu

Name	Possible Values
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 Security Menu

This section refers to the TLS Menu available in the Updater, not Spectralink Software. There is another menu, called TLS Security, available in the Admin menu. Navigate to **Advanced Settings> [password] Administration Settings> TLS Security**. You can modify the following parameters from the TLS Menu:

Table 3-10: TLS Menu

Name	Possible Values
View or install a custom CA cert	URL
A CA certificate that is installed on the handset to be used for TLS authentication.	
View or clear a custom device credentials	Yes, No
A flag to determine whether or not the device certificate can be removed from the handset.	
Configure TLS Profiles	
There are two TLS Platform Profiles and six TLS Application Profiles. See TLS Profile Menu .	
Configure TLS Applications	
See Applications Menu .	

TLS Profile Menu

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

Table 3-11: TLS Profile

Name	Possible Values
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.	

TLS Applications Menu

You can modify which platform profile is used for applications from the Applications Menu. Although not listed here, SIP, browser and LDAP are similarly available:

Table 3-12: 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

Chapter 4: Setting Up the Provisioning Server

This chapter provides instructions for setting up your Spectralink handsets with a central provisioning server. If you are new to this process, it is important to read every section in this chapter. Please also refer to the *Spectralink 84-Series Wireless Telephone Deployment Guide* for basic information.

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

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

Why Use a Provisioning Server?

Spectralink strongly recommends that you use a central provisioning server to install and maintain your Spectralink handsets. 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 handsets, and enables you to store configuration, log, directory, and override files on the server. If you allow the handset write access to your provisioning server, the handset can use the server to upload all of the file types and store administrator and user settings. The handset 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.

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](#).

Provisioning Server Redundancy

You can configure multiple (redundant) provisioning servers—one logical server with multiple addresses—by mapping the provisioning server DNS name to multiple IP addresses. See [Server Redundancy](#) for more information.

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 Spectralink Reseller.

Provisioning Server Security Notes

For organizational purposes, Spectralink recommends configuring a separate log file directory, an override directory, and a contact directory. Each directory can have different access permissions. Normally, LOG, CONTACTS, and OVERRIDES have full read and write access. See [Understanding the Files Written by the Handsets](#) for complete information.

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

Spectralink recommends that you allow file uploads to the provisioning server where the security environment permits. File uploads allow event log files to be uploaded to the provisioning server. File uploads provide backup copies (override configuration files) of changes users make to the handset's configuration settings through the Web server and/or local user interface. These override and log files help service providers and Spectralink provide customer support when diagnosing issues that may occur with the handset operation.

The handset'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 access files in all directories mentioned in the **<MAC-address>.cfg** file. All other files that the handset 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

Spectralink recommends that you use RFC-compliant servers.

Each handset may open multiple connections to the server.

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

If you know the handset 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

A basic 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 Spectralink Software.



Tip: Choosing a Provisioning Protocol

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

See the Spectralink 84-Series Wireless Telephone Deployment Guide for a full explanation of setting up FTP directories for both initial and central provisioning.

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 handset and select **OK**.
For example, *bill123*.
- 6 Select the **Password** checkbox and enter a password.
For example, *1234*. The handset will use this password to log in.



Settings: FTP username and password

When you set up the initial provisioning server as an FTP server, use **administrator** for the username and **admin123** for the password. Ensure all checkboxes are checked.

- 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 handset 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.
IP addresses of your network cards are displayed. One of them (if there are more than one) will be your FTP server.

Downloading Spectralink Software Files to the Provisioning Server

This section explains how to download the Spectralink Software to the provisioning server.



Admin Tip: Upgrading Software

If you need to upgrade software on your handsets, please see Appendix B in the *Spectralink 84-Series Wireless Telephone Deployment Guide*.

Microsoft® Lync® compatibility

Spectralink software is available in two variants – Lync and non-Lync (or open SIP). Starting with Spectralink software 4.3/4.4, even numbered releases support both Lync and open SIP and odd numbered releases support open SIP only. Release 4.7 includes Lync and non-Lync versions which are differentiated by number.

“Non-Lync” versions of the 8440/41/52/53 do not support any Lync capability. A Lync-enabled handset supports Lync telephony, IM, calendaring, and Exchange. A handset without Lync support does not support any Lync functionality including IM, calendaring, and exchange. Handsets cannot be upgraded from non-Lync to Lync-enabled in the field. If customers are unsure if Lync capability will ever be needed, we recommend the purchase of Lync-enabled handsets.

Handsets purchased without Lync capability will not run Lync software releases, e.g. 4.6.x. Handsets with Lync compatibility will run Lync and non-Lync software releases.

<i>Manufacturing date</i>	<i>Support Lync?</i>
Prior to June 2013	Yes
June 2013 and later	2 handset variations: Lync-enabled: supports Lync non-Lync: does not support Lync



Tip

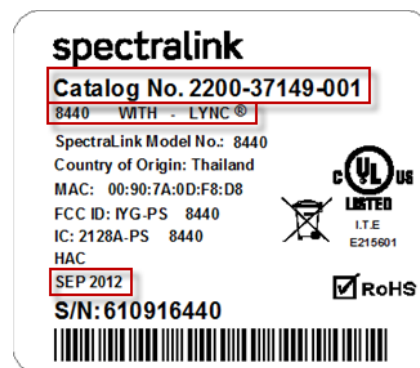
All 84-Series handsets manufactured before June 2013 support Lync.

All handsets have product ID's that identify them as Lync or non-Lync compatible.

84-Series Product IDs with Microsoft Lync Support

Model	Lync SKUs
8440:	2200-37149-001, 2200-37150-001 2200-37174-101, 2200-37175-101
8441:	2200-37290-001, 2200-37290-101
8450:	2200-37152-001, 2200-37153-001 2200-37176-101, 2200-37177-101
8452:	2200-37172-001, 2200-37173-001 2200-37198-101, 2200-37199-101
8453:	2200-37294-001, 2200-37294-101

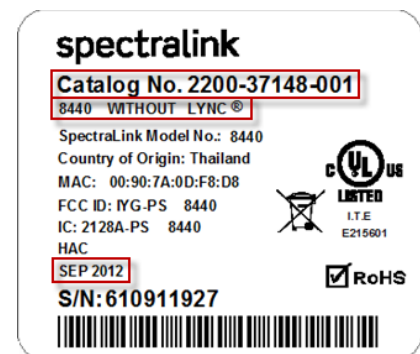
Label example



84-Series Product IDs without Microsoft Lync Support

Model	Open SIP SKUs
8440:	2200-37147-001, 2200-37148-001 2200-37165-101, 2200-37164-101
8441:	2200-37288-001, 2200-37288-101
8450:	All 8450 models support Lync.
8452:	2200-37163-001, 2200-37162-001 2200-37161-101, 2200-37160-101
8453:	2200-37292-001, 2200-37292-101

Label example



Admin Tip

Handsets manufactured prior to June 2013 are not differentiated on the label as to Lync or non-Lync. All handsets manufactured before June 2013 are Lync compatible. Please consult the label date to determine Lync compatibility.

To help understand if your 84-Series handset supports Lync, look at the manufacturing date on the label in the battery compartment. ALL 84-Series handsets produced before June 2013 support Lync. For handsets built during or after June 2013, check the label text. The product ID and the “with Lync” or “without Lync” text on the label will confirm whether or not the handset is Lync-enabled.

Spectralink 84-Series Hardware IDs

Each Spectralink 84-Series model has a unique hardware ID. You can find this number printed on the handset's label inside the battery compartment. This number enables the handset model to identify itself and provides flexibility to the administrator, permitting different models to load different code versions. See [Product Model Number and Hardware ID Mapping](#) for more information about how hardware IDs are used.

<i>Model Name</i>	<i>Hardware ID</i>
SL8440	3111-36150-001
SL8450	3111-36152-001
SL8452	3111-36154-001
SL8441	3111-67360-001
SL8453	3111-67361-001

Go to the [Spectralink Support Website](#) to download current and past releases and access supporting documentation.

Spectralink provides the Spectralink Software download in ZIP file format.

To download the Spectralink Software :

- 1 Access Spectralink Software from the [Spectralink Support Website](#).
- 2 Acknowledge that you read the notices, accept the agreement, and choose **Submit**.
- 3 Save the Spectralink Software ZIP file download.
- 4 Extract (uncompress) the ZIP file.

Copy all files from the distribution ZIP file to working directory on the provisioning server, maintaining the same folder hierarchy. To simplify provisioning, Spectralink recommends, as a best practice, to start creating new configuration files from unedited template files containing the default values. Rename the template file to your specific file name as you configure and add specific parameter values for your site.

- You will see one sip.ld file that includes all handset models. Accompanying folders contain the configuration file templates and localization files.



See the Release Notes for a Description of all Parameters for a Spectralink Software Release

For a description of each file in a Spectralink Software distribution, see the *Spectralink Software Release Notes* for a particular Spectralink Software release on the [Spectralink Software Support Center](#).

Deploying and Updating Spectralink Handsets with a Provisioning Server

This section explains how to deploy and update Spectralink handsets from a provisioning server. If you are provisioning the handsets using a provisioning server for the first time, follow the provisioning process described in the *Spectralink 84-Series Wireless Telephone Deployment Guide*.

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 handset features parameters in another file. Configuration file variances are explained in the *Spectralink 84-Series Wireless Telephone Deployment Guide*.

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](#).

Shortcut Method to Deploy Spectralink Handsets with a Provisioning Server

The following steps are a shortcut method for provisioning procedure. Please use the *Spectralink 84-Series Wireless Telephone Deployment Guide* for detailed instructions for each of these steps.

To deploy handsets with a provisioning server using a shortcut method:

- 1 Obtain a list of MAC addresses for the handsets you want to deploy.
The MAC address is a 12-digit hexadecimal number on a label on the back of the handset and on the outside of the shipping box. It is also available on the **Status** menu.
- 2 Create a per-handset **<MACAddress>-ext.cfg** file.



Do NOT use these names for a per-handset configuration file

Do NOT use the following file names as your per-handset 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 handset itself to store user preferences (overrides) and logging information.

Add handset 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

Some 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 [Time and Date Display](#)). Modify any settings to match your WLAN and network as needed.


4 Create a master configuration file by performing the following steps:

- a Enter the name of each per-handset and per-site configuration files created in steps 2 and 3 in the CONFIG_FILES attribute of the master configuration file (000000000000.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, so that it points to the directory where user profile files are stored.
- 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 handset to point to the IP address of the provisioning server and set up the user:

6 On the handset's Home screen or idle display, select **Settings> Advanced Settings> [enter password]> Administration Settings> Network Configuration> Provisioning Server> DHCP Menu**.

When prompted for the administrative password, enter **456**.

- a Open the **DHCP Server** menu by pressing the **Select** softkey. Set **Boot Server** to **Static** and press  to return to the Provisioning Server menu.
- b Scroll down to Server Type and ensure that it is set to FTP.

- c Scroll down to Server Address and enter the IP address of your provisioning server.
- d Press the Edit softkey to edit the value and the OK softkey to save your changes.
- e Scroll down to Server User and Server Password and enter the user name and password of the profile you created on your provisioning server.
- f 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 softkey twice.
- g Scroll down to Save & Reboot, and then press the Select softkey.

The handset reboots.

After this step, if the file does not exist, the Spectralink Software will try the unmodified APPLICATION APP_FILE_PATH attribute (sip.ld).

For more information, see **Parsing Vendor ID Information**.

- 7 Ensure that the configuration process completed correctly.

On the handset, press the **arrow** key, and then select **Settings> Status> Platform> Application> Main> OK** to see the Spectralink Software version and **Settings> Status> Platform> Configuration> OK** to see the configuration files downloaded to the handset.

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 handset will upload two logs files to the LOG_DIRECTORY directory:

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

You can now test your deployment by making calls and testing features.

Upgrading Spectralink Software

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



Updating Spectralink Software on a Single Phone

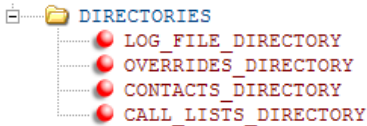
Starting with Spectralink Software 4.3.x, you can use the Software Upgrade tool in the Web Configuration Utility to update the Spectralink Software version running on a single handset. Note that configuration changes made to individual handsets using the Web Configuration Utility will override configuration settings made using central provisioning. For instructions on how to update Spectralink Software, see Technical Bulletin [Using the Software Upgrade Tool in the Web Configuration Utility](#).

To continue setting up a provisioning server, use the instructions in the *Spectralink 84-Series Wireless Telephone Deployment Guide*.

Chapter 5: Understanding the Files Written by the Handsets

The Spectralink 84-Series handsets can write several types of information into directories set up on the central provisioning server. These directories serve as repositories for the different types of files written by the handsets and allow an administrator to easily find needed information. If separate directories are not provisioned, and if the handset has write privileges, all files will be written to the root directory.

The *Spectralink 84-Series Deployment Guide* explains how to set up directories on the central provisioning server (or elsewhere) so that the files written by the handset can be sorted and viewed efficiently. These directories must have full read and write access. The “Directory element” in the table below is referring to the directories that are provisioned in the master configuration file. The Directory names tell the handset where to write the files it produces. For example:

	Directory elements	<div><div>\Log_Files</div><div>\Overrides</div><div>\Contacts</div><div>\Call_Lists</div></div>	Directory names. These are assigned during provisioning of the master configuration file.
---	---------------------------	---	--

Filenames for the files produced by the handset start with the MACaddress and use an extension to signify the type of information in it. E.g. 00907a0cd989-phone.cfg is the filename of an override file created when the user changed a parameter in the handset keypad menus.

If User Profiles are deployed, files start with the login name instead of the MACaddress. E.g. lgates-phone.cfg.

Log files are written and stored as text files. All other files are written and stored as .xml files.

Type of file	Directory element	Contents	Extension
Log	LOG_FILE_DIRECTORY	This is the directory where the handset will write its log files.	-boot.log -app.log
Override	OVERRIDES_DIRECTORY	Parameters set in the .cfg files can be overridden when changed in the Web Configuration Utility or in the handset menus using its keypad. This directory stores these overrides by MACaddress so that they are available when the handset restarts.	-phone.cfg -web.cfg
Contacts	CONTACTS_DIRECTORY	Contacts are stored by MACaddress in this directory as a backup so that they can be reloaded when the handset reboots.	-directory.xml
Calls	CALL_LISTS_DIRECTORY	Call lists are stored by handset or by User Profile.	-calls.xml

Log Files

Log files are discussed in detail in the Troubleshooting section.

Overrides

Certain configuration methods take precedence over other methods and produce override files on the central provisioning server. The handset reads these files when it checks in with the central provisioning server and uses the information stored in them for operational information and parameter configuration.

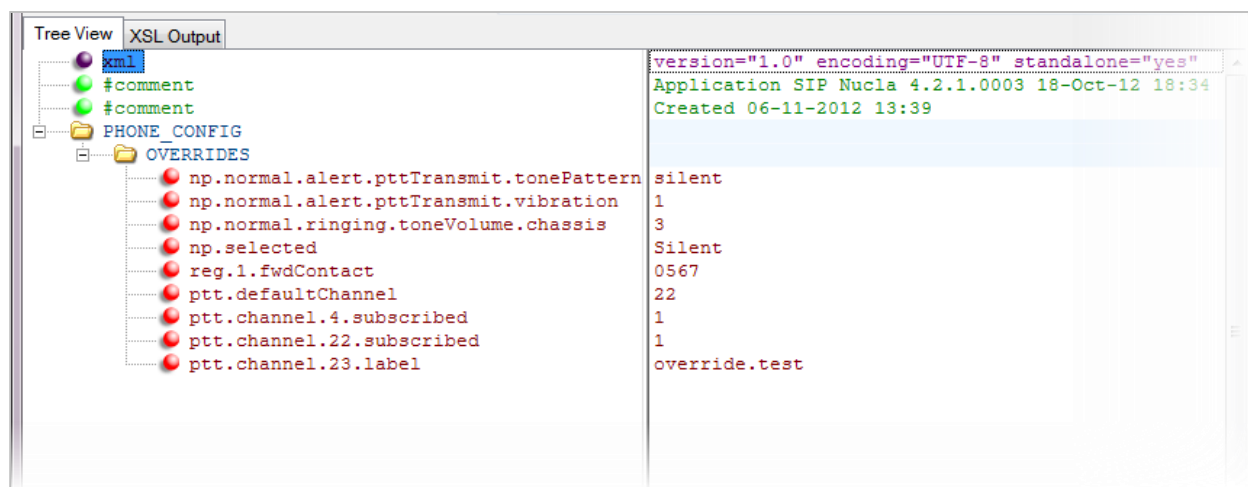
The precedence order for configuration methods follows this sequence (highest to lowest priority):

- 1 Local handset user interface
- 2 Web Configuration Utility
- 3 Central Provisioning Server
- 4 Default values

If you set a parameter in the central provisioning server and the user later changes it in the handset menus, the setting made by the user will take precedence and override the setting in the central provisioning server. This action updates an overrides file that is stored in the Overrides directory, if specified.

Note that although certain parameters take precedence and *override* other parameters, this type of behavior does NOT produce an override file. This behavior is fully explained in the given parameter description. We use the term “override” to describe parameter precedence. We use the term “override” to describe configuration method precedence.

Example:



This example shows an overrides file for one handset that shows what the user has done to change from the default or configured settings:

- set the normal notification profile (np) to a silent ring, etc.
- set the selected notification profile to silent,
- has forwarded a call to extension 0567,
- changed various PTT channel settings.

Contacts

Each handset can store up to 9999 contacts in its local Contact Directory. The Contact Directory can be viewed on the handset by navigating to **Home> Contacts / Call Lists> Contact Directory**. This contact list is also written to the central provisioning server and stored in a Contacts Directory whenever any information is changed. Contact information is also stored locally in the handset so the file on the central provisioning server serves as a backup in case the handset loses its memory.

Call List

The 84-Series handset records missed, received and placed calls in a call list that can be viewed on the handset by navigating to **Home> Contacts / Call Lists> Call Lists**. Call history is stored locally on the handset and will survive a restart or reboot unless the list has been cleared. The user can use the call list to redial previous outgoing calls, return incoming calls, and save contact information from call list entries to the contact directory.

Such calls are logged in the Call List directory that resides on the central provisioning server. These call logs contain call information such as remote party identification, time and date of the call, and call duration. All call logs are enabled by default but can be disabled by turning off the unwanted parameter.

The Call List directory on the central provisioning server is a real-time file. When a new call is made, the call is logged. When the lists are cleared on the handset, the calls in the cleared list are also cleared out of the Call List directory on the central provisioning server.

Table 6-1: Configuring the Call Lists

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
feature.callList.enabled 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

Parameter	Permitted Values	Default
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.		

Call log example

A logged call from the call list looks like this:

callList	
xmlns	<code>http://schema.polycom.com/SIPCallList</code>
xmlns:xsi	<code>http://www.w3.org/2001/XMLSchema-instance</code>
xsi:schemaLocation	<code>http://schema.polycom.com/SIPCallList</code>
call	
direction	Out
disposition	Normal
line	1
protocol	SIP
startTime	2012-11-01T14:00:46
duration	PT0S
count	1
source	
address	7569-A
name	
destination	
address	8884889438
name	Polycom Helpdesk

The next table describes each element and attribute that displays in the call log

Table 6-2: Call Log Elements and Attributes

Element	Permitted Values
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
startTime The start time of the call. For example: 2010-01-05T12:38:05 in local time.	String
duration The duration of the call, beginning when it is connected and ending when the call is terminated. For example: PT1H10M59S.	String
count The number of consecutive missed and abandoned calls from a call destination.	Positive Integer

<i>Element</i>	<i>Permitted Values</i>
destination	Address
<p>The original destination of the call.</p> <p>For outgoing calls, this parameter designates the outgoing call destination; the name is initially supplied by the local handset (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.</p> <p>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 handset).</p>	
source	Address
<p>The source of the call (caller ID from the call recipient's perspective).</p>	
Connection	Address
<p>An array of connected parties in chronological order.</p> <p>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.</p>	
finalDestination	Address
<p>The final connected party of a call that has been forwarded or transferred to a third party.</p>	

Part III: Configuring Features

Part III provides you with an in-depth look at advanced features and parameters. The *Spectralink 84-Series Wireless Telephone Deployment Guide* covers parameters that are used by 80% of all installations. However, since requirements vary, Part III covers the entire range of configurable parameters for every type of feature and functional requirement.

Part III is arranged by types of features and settings that you might need to deploy to further customize your installation. This Part is divided into four Chapters that cover:

- An explanation of inherent features that are not configurable,
- User settings that can be set by the administrator but the user can freely alter through the user menus. Some of these options can be locked or made unavailable by the administrator.
- Feature settings that are configured by the administrator. These settings are optional and only available if configured by an administrator. In some cases once the feature is configured, user menus contain options that the user can configure to customize the feature for individual use and preferences.
- System settings that the administrator configures to adjust the way the handset interacts with the greater infrastructure.

Configuration parameters are named for the function they provide but frequently several types of parameters must be configured to provision a feature or function. Therefore this section is arranged by feature and function, not by parameter name.

The parameters listed in this section are available in the Config folder that is downloaded with the software. For any parameters not available in the scenario templates, look in the Troubleshooting folder for the “everything.cfg” file. You will find an alphabetical folder hierarchy of all the parameters detailed in this document. To find the exact parameters you want to use, open the file and use your search tools.

The easiest way to provision any parameter is to simply drag and drop or copy/paste it from the source template into the .cfg file you will use to deploy the handsets. Edit it in your final .cfg file per the instructions in the parameter list.

Chapter 6: Features that Cannot be Configured

Audio Processing Features

The Spectralink 84-Series handsets have these 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-handset 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 acoustic echo cancellation (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. This feature is different from Voice Activity Detection.

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 handset 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 handset is designed to minimize latency for audio packet transmission.

Call Timer

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

There are no related configuration changes.

Called Party Identification

By default, the handset displays and logs the identity of parties called from the handset. The handset obtains called party identity from the network signaling. Because Called Party Identification is a default state, the handset 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.

Connected Party Identification

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

Microphone Mute

The handsets have a microphone mute softkey. When you activate microphone mute, a mute icon will display on the status bar.

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

Synthesized Call Progress Tones

Spectralink handsets 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 handset match standard North American tones.

Chapter 7: Configurable Features on the User Menus

Features that are available on the User menus, such as ring tones, can be configured by the administrator in the central provisioning server but may be changed by the user, creating override files stored on the central provisioning server. Some user-level parameters that are made available to users by default can be disabled by the administrator if certain usability factors need to be enforced. This chapter covers features that are available on the User menus that have user-level parameters that can be configured on the central provisioning server.

Some features, such as Push-to-talk, are set up by the administrator and are designed to be customized by the user, like subscribing to certain channels that have been enabled by the administrator. These features are covered in the next chapter.

Features that are not on the User menus but can be set through the Web Configuration Utility are covered in the next chapter.

Call Forwarding

The handset provides a flexible call forwarding feature that enables you to forward incoming calls to another destination. To enable and set call forwarding from the handset, navigate to **Home> Settings> Feature Settings> Forward**. From this menu you can apply call forwarding in the following ways:

- To all calls
- When your handset is busy
(Since the 84-Series handsets support 24 incoming calls by default [call.callsPerLineKey] they won't offer a busy signal to the PBX unless it's handling 24 calls to any given handset, an unlikely event. In order to forward on busy, the callsPerLineKey must be set to 1 or 2 instead of 24.)
- When the handset has not been answered within a specified number of rings

Call forwarding is also available as a divert function in the Contacts Directory:

- To incoming calls from a specific caller or extension
- You can have incoming calls forwarded automatically to a predefined destination you choose or you can manually forward calls to a destination.

To enable server-based call forwarding, see your PBX manual and the Interop Guide relating to that server.



Troubleshooting: If Call Forwarding Does Not Work

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 handset, 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 handset of a call's history. For example, when you enable call forwarding, the Diversion header allows the receiving handset to indicate who the call was from, and the handset number it was forwarded from.

Summary

<i>Parameter</i>	<i>Used to:</i>
<code>volpProt.SIP.serverFeatureControl.cf</code>	Enable or disable server-based call forwarding
<code>volpProt.SIP.serverFeatureControl.localProcessing.cf</code>	Enable or disable local call forwarding behavior when server-based call forwarding is enabled
<code>volpProt.SIP.header.diversion.*</code>	Enable or disable the display of the Diversion header and the order in which to display the caller ID and number
<code>divert.*</code>	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.
<code>reg.x.fwd.*</code>	Enable or disable server-based call forwarding as a per-registration feature

Table 7-1: Configuring Call Forwarding

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
<code>volpProt.SIP.serverFeatureControl.cf</code>¹	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.		
<code>volpProt.SIP.serverFeatureControl.localProcessing.cf</code>	0 or 1	1
If set to 0 and <code>volpProt.SIP.serverFeatureControl.cf</code> is set to 1, the handset will not perform local Call Forward behavior. If set to 1, the handset will perform local Call Forward behavior on all calls received.		
<code>volpProt.SIP.header.diversion.enable</code>¹	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.		
<code>volpProt.SIP.header.diversion.list.useFirst</code>¹	0 or 1	1
If set to 1, the first diversion header is displayed. If set to 0, the last diversion header is displayed.		
<code>reg.x.fwdcontact</code>	string	Null
The forward-to contact for calls when always is specified. If Null, calls are not forwarded.		
<code>reg.x.fwd.noanswer.status</code>	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> .		

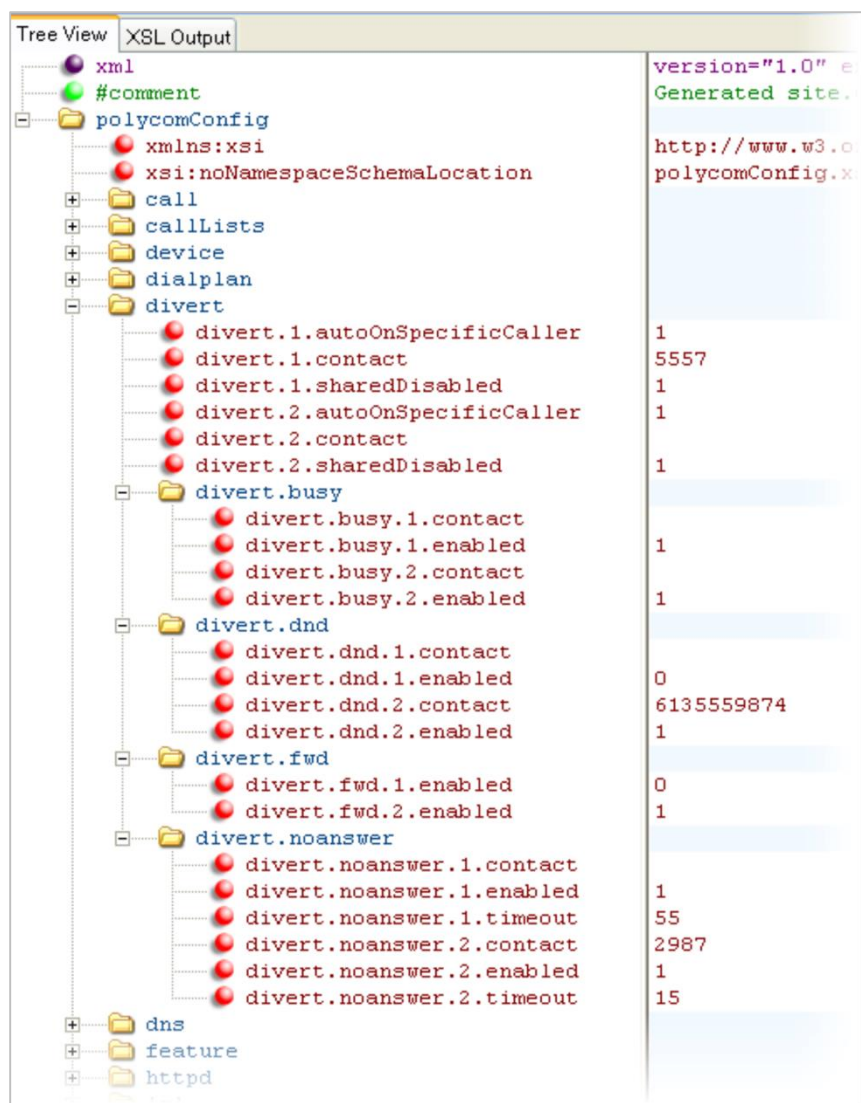
<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
reg.x.fwd.noanswer.contact The forward-to contact used for calls forwarded due to no answer. If Null, the contact specified by <code>divert.x.contact</code> will be used.	string	Null
reg.x.fwd.noanswer.ringCount The number of seconds the handset 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.	0 to 65535	0
reg.x.fwd.busy.status 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> .	0 or 1	0
reg.x.fwd.busy.contact 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.	string	Null

¹ Change causes handset to restart or reboot.

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 overrule `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 handset.
- 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 handset.
- 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 handset, 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`.



The handset 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. SL8400: x=1-6.

Table 7-2: Call Diversion (Call Forwarding) Parameters

Parameter	Permitted Values	Default
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@Spectralink.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 <i>busy</i> , <i>dnd</i> , and <i>noAnswer</i> 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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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.		
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 x 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 x. 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 x). If 1, a <i>Forward</i> softkey will display on the flyout menu when you press the <i>Features</i> softkey.		
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 handset to restart or reboot.

² Change causes handset to restart or reboot. If server-based call forwarding is enabled, this parameter is disabled.

Keypad Lock

Keypad Lock is enabled by default. This feature locks the keypad to prevent inadvertent dialing while the phone is idle and/or while in a call. The feature adds a **Keypad Lock** and/or an **In-call Keypad Lock** option on the Features softkey menu. Once the handset is locked for idle, an **Unlock** softkey appears. When pressed it will unlock the keypad. No password is required. The in-call keypad lock will expire when the phone call is ended.

If your deployment requires additional handset security, please see the **Phone Lock** section.

Table 7-3: Keypad Lock

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
keypadLock.enabled¹	0 or 1	0
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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
inCall.keypadLock.enabled	0 or 1	0
If enabled the In-call Keypad Lock item will appear on the Profile soft key when a call is connected. Note that if more than one call connected at the same time, this item will not show up as the space on the flyout is used by other items.		
inCall.keypadLock.autoLockTime	0 to 30	0
If 0, auto-lock is not enabled. If not zero then the In-call Keypad Lock will be engaged nn seconds after a call is fully connected. Only 1 call can engage the lock. If a second call is started during the timeout period for the first call, the auto-lock for the second call will be ignored. The In-call Keypad Lock item on the Profile soft key flyout can still be used even if auto-lock is configured.		

¹ Change causes handset to restart or reboot.

Multi Key Answer

Multikey Answer enables you to answer incoming calls by pressing any key on the handset's keypad. Multikey Answer is disabled by default but may be enabled by the administrator. You cannot use the Multi Key Answer feature for Open Application Interface (OAI) calls, Group Paging, or Push-to-talk (PTT) calls. Navigate to the option by going to **Settings> Basic Settings> Preferences> Multi Key Answer**.

Table 7-4: Enabling Multi Key Answer

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
up.multiKeyAnswerEnabled	0 or 1	0
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.		

¹ Change causes handset to restart or reboot.

Notification Profiles

The Spectralink handsets support four profiles for notification alerts: **Normal**, **Silent**, **Meeting**, and **Custom1**. You can customize each profile with a unique name, unique ringtones, alerts, and vibrations for specific situations. For example, you can customize barcode scan alerts or when you receive an instant message.

Notification Profiles are selectable on the handset by selecting a profile from the standby mode Profile softkey or by navigating to **Settings> Basic Settings> Notification Profiles** where profiles can be modified by the user.

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 handset will maintain changes you make to the ringer volume when the handset reboots or restarts.

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. 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**.

Configure the default notification profile by setting the parameter shown in the following table:

Table 7-5: Notification Profile Selection Parameter

Parameter	Permitted Values	Default
np.selected	Normal, Silent, Meeting, Custom1	Normal
The initial profile that is selected when the handset 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 handset powers on the next time.		

Table 7-6: Normal Profile Alert Parameters

Parameter	Permitted Values	Default
np.normal.label	String	Normal
The name of the profile type.		
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 when 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 when the handset is undocked.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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.		
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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
np.normal.ringing.oai1.vibration	off, continuous, shortPulse, longPulse	off
The vibration pattern for Open Application Interface (OAI) communications.		
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 2013 or 2010.		
np.normal.ringing.privateLine.vibration	off, continuous, shortPulse, longPulse	shortPulse
The vibration pattern for a private line registered to Microsoft Lync Server 2013 or 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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
np.normal.ringing.toneVolume.dock	-1000 to 1000	-21
The attribute is set (on adjusting ring volume) when handset is at the speakerphone dock and Normal profile is active. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		

Table 7-7: 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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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.		
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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
np.silent.ringing.calls.vibration	off, continuous, shortPulse, longPulse	off
The ringtone (see se.pat.ringer) and vibration (1 to enable) for normal calls.		
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 2013 or 2010.		
np.silent.ringing.privateLine.vibration	off, continuous, shortPulse, longPulse	shortPulse
The vibration pattern for a private line registered to Microsoft Lync Server 2013 or 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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
np.silent.ringing.toneVolume.dock	-1000 to 1000	-21
The attribute is set (on adjusting ring volume) when handset is at the speakerphone dock and Silent profile is active. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		

Table 7-8: Meeting Profile Alert Parameters

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
np.meeting.label The name of the profile type.	String	Meeting
np.meeting.alert.barcodeBeep.tonePattern np.meeting.alert.barcodeBeep.vibration The tone pattern and vibration (1 to enable) for the alert played when a barcode is scanned.	Any tone (see se.pat.misc) 0 or 1	misc2 0
np.meeting.alert.docked.tonePattern np.meeting.alert.docked.vibration The tone pattern and vibration (1 to enable) for the alert played if the handset is docked.	Any tone (see se.pat.misc) 0 or 1	positiveConfirm 0
np.meeting.alert.undocked.tonePattern np.meeting.alert.undocked.vibration The tone pattern and vibration (1 to enable) for the alert played if the handset is undocked.	Any tone (see se.pat.misc) 0 or 1	negativeConfirm 0
np.meeting.alert.instantMessaging.tonePattern np.meeting.alert.instantMessaging.vibration The tone pattern and vibration (1 to enable) for the instant message alert.	Any tone (see se.pat.misc) 0 or 1	instantMessage 0
np.meeting.alert.localHoldNotification.tonePattern np.meeting.alert.localHoldNotification.vibration The tone pattern and vibration (1 to enable) for the local hold notification alert.	Any tone (see se.pat.misc) 0 or 1	localHoldNotification 0
np.meeting.alert.lossOfNetwork.tonePattern np.meeting.alert.lossOfNetwork.vibration The tone pattern and vibration (1 to enable) for the alert played if the network is lost.	Any tone (see se.pat.misc) 0 or 1	misc1 0
np.meeting.alert.lowBattery.tonePattern np.meeting.alert.lowBattery.vibration The tone pattern and vibration (1 to enable) for the alert played if the battery is low.	Any tone (see se.pat.misc) 0 or 1	misc1 0
np.meeting.alert.veryLowBattery.tonePattern np.meeting.alert.veryLowBattery.vibration The tone pattern and vibration (1 to enable) for the alert played if the battery is very low.	Any tone (see se.pat.misc) 0 or 1	misc1 0
np.meeting.alert.messageWaiting.tonePattern np.meeting.alert.messageWaiting.vibration The tone pattern and vibration (1 to enable) for the alert played if there is a message waiting.	Any tone (see se.pat.misc) 0 or 1	messageWaiting 0
np.meeting.alert.negativeConfirm.tonePattern np.meeting.alert.negativeConfirm.vibration The tone pattern and vibration (1 to enable) for the negative confirmation alert.	Any tone (see se.pat.misc) 0 or 1	negativeConfirm 0
np.meeting.alert.positiveConfirm.tonePattern np.meeting.alert.positiveConfirm.vibration The tone pattern and vibration (1 to enable) for the positive confirmation alert.	Any tone (see se.pat.misc) 0 or 1	positiveConfirm 0
np.meeting.alert.pttTransmit.tonePattern np.meeting.alert.pttTransmit.vibration The tone pattern and vibration (1 to enable) for the alert played if sending a push-to-talk page.	Any tone (see se.pat.misc) 0 or 1	misc3 0
np.meeting.alert.pttWait.tonePattern np.meeting.alert.pttWait.vibration The tone pattern and vibration (1 to enable) for the push-to-talk wait alert.	Any tone (see se.pat.misc) 0 or 1	misc4 0

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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.		
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 2013 or 2010.		
np.meeting.ringing.privateLine.vibration	off, continuous, shortPulse, longPulse	shortPulse
The vibration pattern for a private line registered to Microsoft Lync Server 2013 or 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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
np.meeting.ringing.toneVolume.dock	-1000 to 1000	-21
The attribute is set (on adjusting ring volume) when handset is at the speakerphone dock and Meeting profile is active. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
np.meeting.ringing.toneVolume.reserved	-1000 to 1000	-21
Not currently used. Reserved for future use.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		

Table 7-9: 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	positiveConfirm 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.		
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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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.		
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 2013 or 2010.		
np.custom1.ringing.privateLine.vibration	off, continuous, shortPulse, longPulse	shortPulse
The vibration pattern for a private line registered to Microsoft Lync Server 2013 or 2010.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
np.custom1.ringing.toneVolume.dock	-1000 to 1000	-21
The attribute is set (on adjusting ring volume) when handset is at the speakerphone dock and Custom1 profile is active. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		
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. Although the permitted values are -1000 to 1000, the practical limits used by the handset are -50 to 10.		

Time and Date Display

A clock and calendar are enabled by default. View/edit by navigating to **Home> Settings> Basic Settings> Preferences> Time & Date**. 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 handset is in certain modes. For example, the display format can change when the handset goes from idle mode to an active call.

There are multiple formats to this parameter and as a result there is a hierarchy of precedence. Example: the device.snntp parameter would be considered a base level parameter whereas the tcpipApp.snntp.address parameter takes precedence over the device.snntp parameter. Further still DHCP can override both and take precedence. However, tcpipApp.snntp.address.overrideDHCP can be configured to override DHCP.

The templates contain the parameter device.snntp in the wireless.cfg file. No other template file contains any of these snntp base parameters. They are available in the everything.cfg.

In order to display the time and date accurately, you will have to synchronize the handset to the Simple Network Time Protocol (SNTP) time server. Until a successful SNTP response is received, the handset will continuously flash a time and date to indicate that they are not accurate.



Caution: Some language parameters may overrule lcl.dateline parameters.

Certain languages use date and time settings that overrule lcl.dateline parameters. See the Languages section for details.

Table 7-10: Setting the Time and Date Display

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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>
lcl.datetime.date.format	string which includes 'D', 'd' and 'M' and two optional commas	D,Md
Controls format of date string. 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.datetime.date.longFormat	0 or 1	1
If set to 1, display the day and month in long format (Friday/November), otherwise, use abbreviations (Fri/Nov).		
lcl.datetime.time.24HourClock	0 or 1	0
If set to 1, display time in 24-hour clock mode rather than a.m./p.m.		

Table 7-11: Date Formats

<i>lcl.datetime.date.format</i>	<i>lcl.datetime.date.longformat</i>	<i>Date Displayed on Phone</i>
dM,D	0	19 Aug, Fri
dM,D	1	19 August, Friday
Md,D	0	Aug 19, Fri
Md,D	1	August 19, Friday
D,dM	0	Fri, 19 Aug
D,dM	1	Friday, 19 August
D,Md	0	Fri, Aug 19
D,Md	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	2011/08/19

Synchronizing with SNTP

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 7-12: Simple Network Time Protocol (SNTP) Parameters

Parameter	Permitted Values	Default
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 overrule 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.		
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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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 handset resynchronizes with the SNTP server. <i>Note:</i> 86400 seconds is 24 hours.		

User Preferences Parameters

Options on various user menus can be pre-configured by the administrator.

Table 7-13: User Preferences Parameters

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
up.backlight.idleIntensity	0, 1, 2, or 3	0
The brightness of the LCD backlight when the handset 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. When the phone is in the charger, this parameter does not take effect, the backlight is always on.		
up.backlight.onIntensity	0, 1, 2, or 3	3
The brightness of the LCD backlight when the handset is active (in use). 0: off, 1 – low, 2 – medium, 3 – high		
up.backlight.timeout	5 to 60	10
The number of seconds to wait before the backlight dims from the active intensity to the idle intensity.		
up.hearingAidCompatibility.enabled	0 or 1	0
If set to 1, the handset audio Rx (receive) equalization is disabled for hearing aid compatibility. If 0, audio Rx equalization is enabled.		
up.idleTimeout¹	0 to 65535, seconds	40
The number of seconds that the handset can be idle before automatically leaving a menu and showing the idle display. If 0, there is no timeout and the handset does not automatically exit to the idle display.		
up.numberFirstCID¹	0 or 1	0
If 0, the caller ID display will show the caller's name first. If 1, the caller's handset number will be shown first.		
up.onHookDialingEnabled	0 or 1	1
If 0, on hook dialing is disabled. If 1, on-hook dialing is enabled.		
up.screenCapture.enabled¹	0 or 1	0
If 0, screen captures are disabled. If 1, the user can enable screen captures from the Screen Capture menu on the handset. <i>Note:</i> when the handset reboots, screen captures are disabled from the Screen Capture menu on the handset.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
up.simplifiedSipCallInfo	0 or 1	0
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.		
up.warningLevel¹	0 to 2	0
If 0, the handset's warning icon and a pop-up message display on the handset for all warnings. If 1, the warning icon and pop-up messages are only shown for critical warnings. If 2, no warnings are displayed. <i>Note:</i> All warnings are listed in the Warnings menu (navigate to Menu> Status> Diagnostics> Warnings on the handset).		
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 handset reboots.		
up.welcomeSoundOnWarmBootEnabled¹	0 or 1	0
If 0, the welcome sound is played when the handset powers up (cold boot), but not after it restarts or reboots (warm boot). If 1, the welcome sound plays each time the handset powers up, reboots, or restarts.		

¹ Change causes handset to restart or reboot.

Chapter 8: Features Configured by the Administrator

Certain features are entirely set by the administrator and for the most part the user cannot change them. However, some have user-configurable options available on the user menus. Features of this type are covered in this chapter.



Finding the Parameters in the Config files

The parameters detailed in this chapter are mostly found in the “everything.cfg” file located in the troubleshooting folder in the Config folder that you download with the software starting with Spectralink software version 4.2.x.

AutoComplete List

The autocomplete list displays when the user goes off hook and when transferring a call. It is composed of your Call List and Contact Directory entries, sorted alphabetically.

When the autocomplete list is presented, the user can type in numbers or letters (using the Dial mode softkey to switch between modes) to get a list of matches.

The default method of finding matches is to match the entered characters to the starting characters in the fields checked following this logic:

- For the Call List entries, the handset checks for matches within the name and the contact fields.
- For the Contact Directory items, the handset checks for matches within the first name, last name, and contact fields.

The search is always case insensitive.

The administrator can configure the handset to have the search find matches that “contain” the entered characters.

Parameter	Permitted Values	Default
autoComplete.useContainsSearch	0 or 1	0
When set to 0, the search will compare the entered characters against the starting characters of each field searched. This is the default behavior. When set to 1, the search will look for the entered characters anywhere inside each field searched.		

Note that the handset limits the search to a certain period of time (because otherwise it can interfere with entering letters in the search field) so not all possible matches may be shown in the list after the user enters a single letter or number. This is especially true when using a “contains” search. The user might need to enter 2 or 3 characters to narrow down the search

enough for the handset to show all the possible matches. This should only be necessary when very large lists are present.

Audio Settings

Context Sensitive Volume Control

The parameters shown below enable you to adjust the volume of handset 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.

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

<voice.volume/>

Table 8-1: Context Sensitive Volume Control

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
voice.volume.persist.bluetooth.headset¹	0 or 1	0
If 0, the Bluetooth headset 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.		
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.		
voice.volume.persist.headset¹	0 or 1	0
If 0, the headset 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.		
voice.volume.persist.handsfree¹	0 or 1	1
If 0, the speakerphone 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.		
voice.volume.persist.usb.handsfree¹	0 or 1	1
If 0, the USB headset 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.		
voice.volume.persist.usbHeadset¹	0 or 1	0
If 0, the USB headset 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.		

¹ Change causes handset to restart or reboot.

<voice/>

The <voice/> parameter controls the settings related to the audio on the handset.

Table 8-2: Voice Parameters

Parameter	Permitted Values	Default
voice.txPacketDelay¹	low, normal, Null	Null
If set to <i>normal</i> or Null, no audio parameters are changed. If set to low and there are no precedence conflicts, the following changes are made: <ul style="list-style-type: none"> • voice.codecPref.G722="1" • voice.codecPref.G711_Mu="2" • voice.codecPref.G711_A="3" • voice.codecPref.<OtherCodecs>="" • voice.audioProfile.G722.payloadSize="10" • voice.audioProfile.G711Mu.payloadSize= "10" • voice.audioProfile.G711A.payloadSize= "10" • voice.aec.hs.enable="0" • voice.ns.hs.enable="0" 		
voice.txPacketFilter¹	0 or 1	0
If 0, no Tx filtering is performed. If 1, narrowband Tx high pass filter is enabled.		

¹ Change causes handset 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.



Caution: Do not change these settings.

Changing any rxQoS settings can impair system operation. Do not change any rxQoS parameters without prior consultation with Spectralink Technical Support.

Table 8-3: Voice Jitter Buffer Parameters

Parameter	Permitted Values	Default
voice.rxQoS.avgJitter¹	0 to 80	20
voice.rxQoS.maxJitter¹	0 to 200	160
The average and maximum jitter in milliseconds for wired network interface voice traffic. avgJitter – The wired interface minimum depth will be automatically configured to adaptively handle this level of continuous jitter without packet loss. maxJitter – The wired interface jitter buffer maximum depth will be automatically configured to handle this level of intermittent jitter without packet loss. Actual jitter above the average but below the maximum may result in delayed audio play out while the jitter buffer adapts, but no packets will be lost. Actual jitter above the maximum value will always result in packet loss. Note that 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.		

Parameter	Permitted Values	Default
voice.rxQoS.wireless.avgJitter¹	0 to 200	70
voice.rxQoS.wireless.maxJitter¹	20 to 500	300
<p>The average and maximum jitter in milliseconds for wireless network interface voice traffic.</p> <p>avgJitter – The wireless interface minimum depth will be automatically configured to adaptively handle this level of continuous jitter without packet loss.</p> <p>maxJitter – 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>Note: if legacy <code>voice.audioProfile.x.jitterBuffer.*</code> parameters are explicitly specified, they will be used to configure the jitter buffer and these <code>voice.rxQoS</code> parameters will be ignored for wireless interfaces.</p>		
voice.rxQoS.ptt.avgJitter¹	0 to 200	150
voice.rxQoS.ptt.maxJitter¹	20 to 500	480
<p>The average and maximum jitter in milliseconds for IP multicast voice traffic (wired or wireless).</p> <p>avgJitter – The PTT/Paging interface minimum depth will be automatically configured to adaptively handle this level of continuous jitter without packet loss.</p> <p>maxJitter – The PTT/Paging 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>Note: if legacy <code>voice.audioProfile.x.jitterBuffer.*</code> parameters are explicitly specified, they will be used to configure the jitter buffer and these <code>voice.rxQoS</code> parameters will be ignored for PTT/Paging interface interfaces.</p>		

¹ Change causes handset to restart or reboot.

Automatic Off-Hook Call Placement

You can configure the handset to automatically place a call to a specified number when you go off-hook. This feature is sometimes referred to as *Hot Dialing*. The handset goes off-hook when you lift the handset, press the New Call softkey, or press the headset or speakerphone buttons on the handset. You can specify an off-hook call contact and enable or disable the feature for specific line registrations.

Table 8-4: Enabling Automatic Off-Hook Call Placement

Parameter	Permitted Values	Default
call.autoOffHook.x.enabled¹	0 or 1	0
<p>Enable or disable the feature</p> <p>If <code>enabled</code> is set to 0, no call is placed automatically when the handset 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>.</p>		
call.autoOffHook.x.contact¹	a SIP URL	Null
<p>The contact address to where the call is placed</p> <p>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@Spectralink.com</code>).</p>		

¹ Change causes handset to restart or reboot.

Background Images

The Spectralink 84-Series wireless handsets include a feature that enables you to add a custom background using a digital image. Using a digital image enables you to display a company logo or product brand as the background on your handset. Supported graphics files are PNG, JPEG, or BMP images. The maximum supported size is 240x270 pixels. Progressive or multiscan JPEG images are not supported.

After one or more backgrounds have been provisioned, the user can select one of them by navigating to **Menu> Settings> Basic> Preferences> Background**.



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 handset screen. As a general rule, backgrounds should be light in shading for better handset and feature usability.

Configuring Background Images

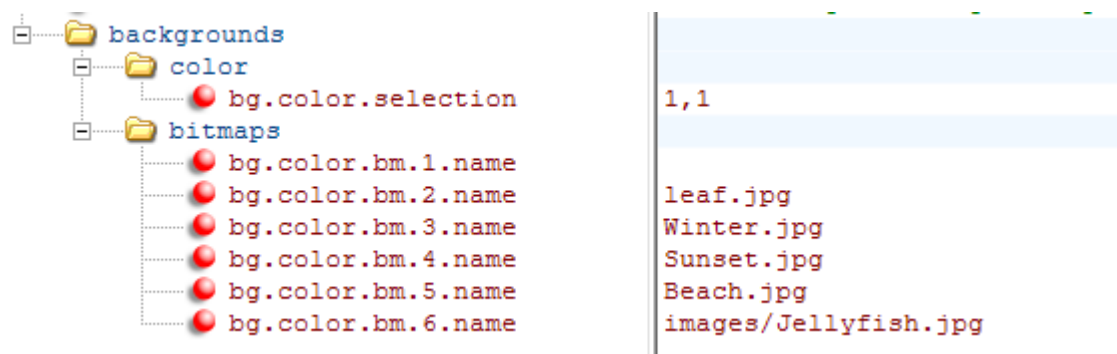
You may want to define a set of images that the user can select from and set one of them as the default. This feature is commonly used to set the company logo as a background or to distinguish handsets that are provisioned for specific purposes or in specific groups. Limit the configurable options by limiting the number of images available.

Table 8-5: Background Parameters

Parameter	Permitted Values	Default
bg.color.selection	w,x	1,1
Set the background. Specify which type of background (w) and index (x) for that type is selected on reboot. The default selection is 1,1 the first solid background. Use w=1 and x=1 (1,1) to select the built-in image. Use w=3 and x= 1 to 6 to select one of the six background <i>bm</i> images		
bg.color.bm.x.name Phone screen background image file	URL or file path of a BMP or JPEG image	null
The name of the image file (including extension).		

Example configuration

In this example, four of the graphic images are located in the root directory of the central provisioning server. The fifth image, Jellyfish.jpg, is located in a subdirectory. Each of these images will appear on the Background menu and is user-selectable.



The figure shown next is an example of a digital image background on a Spectralink 84-Series handset.



Feature and Basic Settings Menu Password

Certain installations need to restrict access to options on the Settings menu. The Advanced Settings option is already behind an admin password. These parameters allow you to require a password to access Basic Settings and Feature Settings.



<i>Parameter</i>	<i>Permitted values</i>	<i>Default</i>
settingsLock.basicSettingsPassword	String (1-32 characters)	null
If set, the indicated password is required to enter the Basic Settings menu. Also causes Edit item to be removed from the Profiles softkey flyout menu. This allows user to change which profile is current, but not modify the settings for each profile. Defaults to Null which means no password required.		
settingsLock.featureSettingsPassword	String (1-32 characters)	Null
If set, the indicated password is required to enter the Feature Settings menu. Also causes the Forward item to be removed from the Features softkey flyout menu. Defaults to Null which means no password required. Note that the Feature Settings menu has DND, Forward and also the Microsoft Lync Signin/Signout menus. However, the Lync Signin/Signout items can be configured to be on the Features softkey flyout menu so one can always signin and signout.		
settingsLock.disallowProfileSoftkey	0 or 1	0
If 1, the Profile softkey will not be shown at all. Default is 0 which is current behavior.		

Call Hold

The 84-Series handset has local hold functionality. Rarely, a server may not recognize this functionality and the call can be lost. If this situation occurs, a server-based call hold feature may be required and the parameters described in this section can be used.

The purpose of call hold is to pause activity on one call so that you can use the handset for another task, for example, to place or receive another call or to search your handset's menu for information. 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.

Table 8-6: Enabling Call Hold

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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 handset 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 handset 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 handset will send a reinvite with a stream mode parameter of “inactive” when a call is put on hold.</p> <p>NOTE: The handset 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>		
call.hold.localReminder.enabled¹	0 or 1	0
<p>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.</p>		
call.hold.localReminder.period¹	non-negative integer	60
<p>Specify the time in seconds between subsequent hold reminders.</p>		
call.hold.localReminder.startDelay¹	non-negative integer	90
<p>Specify a time in seconds to wait before the initial hold reminder.</p>		
volpProt.SIP.musicOnHold.uri	a SIP URI	Null
<p>A URI that provides the media stream to play for the remote party on hold. This parameter is used if reg.x.musicOnHold.uri is Null.</p> <p>Note: The SIP URI parameter transport is supported when configured with the values of UDP, TCP, or TLS.</p>		

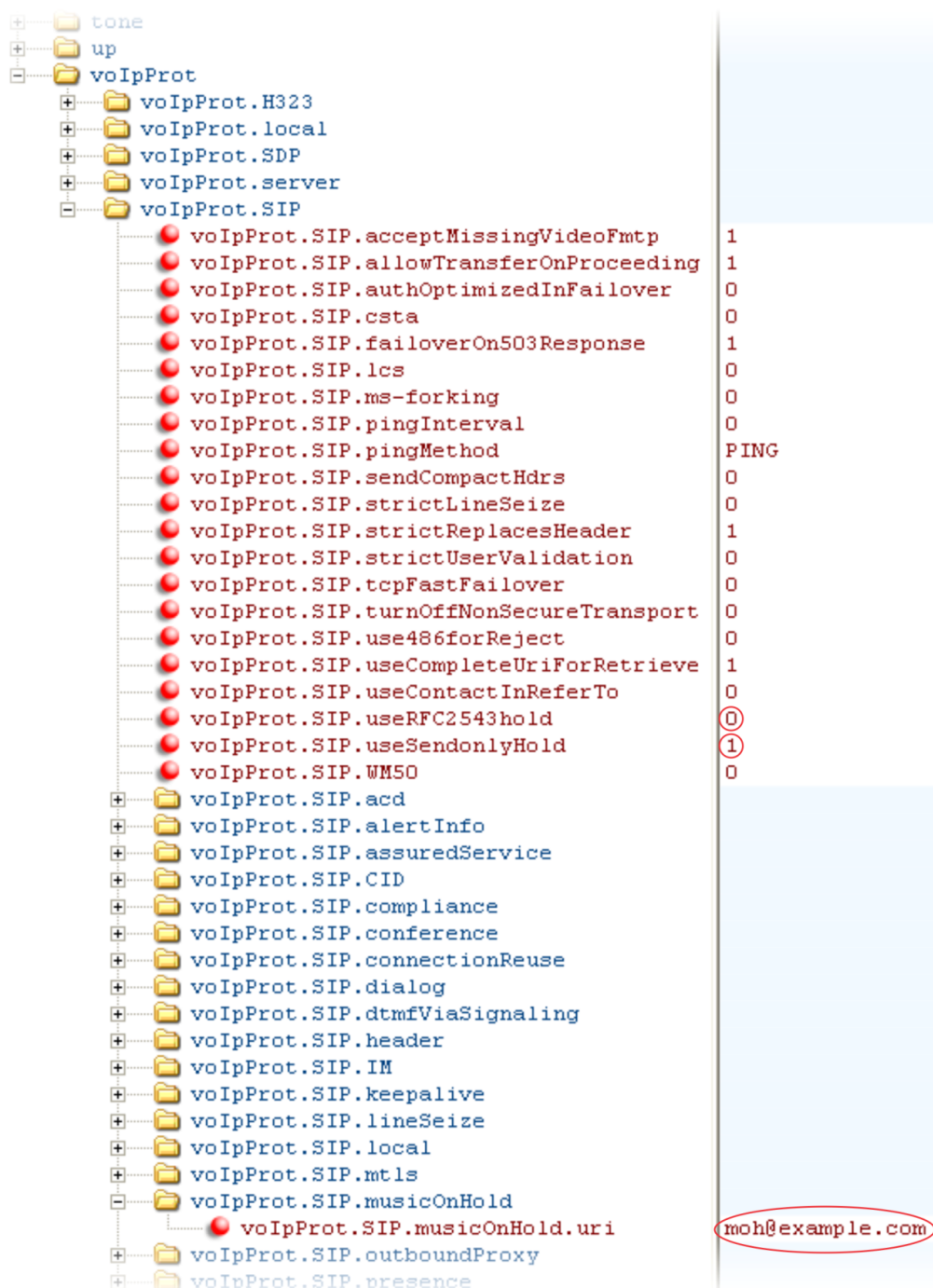
¹ Change causes handset to restart or reboot.

Example Call Hold Configuration

The following two illustrations show a sample configuration for the call hold feature. 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.

xsi:noNamespaceSchemaLocation		polycom
call		
call.dialtoneTimeOut		60
call.directedCallPickupMethod		
call.directedCallPickupString		*97
call.enableOnNotRegistered		1
call.lastCallReturnString		*69
call.localConferenceCallHold		0
call.localConferenceEnabled		1
call.offeringTimeOut		60
call.parkedCallRetrieveMethod		
call.parkedCallRetrieveString		
call.rejectBusyOnDnd		1
call.ringBackTimeOut		60
call.singleKeyPressConference		0
call.stickyAutoLineSeize		0
call.urlModeDialing		0
call.advancedMissedCalls		
call.autoRouting		
call.callWaiting		
call.clickToDial		
call.hold		
call.hold.localReminder		
call.hold.localReminder.enabled		1
call.hold.localReminder.period		30
call.hold.localReminder.startDelay		45
call.hold.remoteNotification		
call.shared		

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`.



Call Handling Features

The call handling features described in this section require support from a SIP server and setup of these features depend 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.

The Spectralink 84-Series handsets can bypass some SIP server requirements and implement various call handling features using Enhanced Feature Keys (EFK) instead. Please refer to the EFK section for more information.

Call Park and Retrieve

You can park an active call and retrieve parked calls from any handset. 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 handset.

Table 8-7: Configuring Call Park and Retrieve

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
feature.callPark.enabled¹	0 or 1	0

If 0, the call park and call retrieve features are disabled. If 1, the features are enabled.

¹ Change causes handset to restart or reboot.

Call Waiting Alerts

By default, the handset will alert you to incoming calls while you are in an active call.

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
call.callWaiting.enable	0 or 1	1
If 1, the handset 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.		
call.callWaiting.ring¹	beep, ring, silent	beep
Specifies the ringtone of incoming calls when another call is active. If set to Null, the default value is beep.		

¹ Change causes handset to restart or reboot.

Calling Party Identification

By default, the handset displays the identity of incoming callers if available to the handset through the network signal. If the caller is in the contact directory, you can choose to display that name instead. Note that the handset cannot match the identity of calling parties to entries in the Corporate Directory.

Table 8-8: Configuring Calling Party Identification

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
up.useDirectoryNames¹	0 or 1	1
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.		

¹ Change causes handset to restart or reboot.

Missed Call Notification

By default, the missed call notification displays on the 84-Series handset's status bar. A counter shows the number of missed calls. The counter is reset by viewing the Missed Calls list on the handset.

The Missed Call Notification can be disabled for each registered line on a handset.

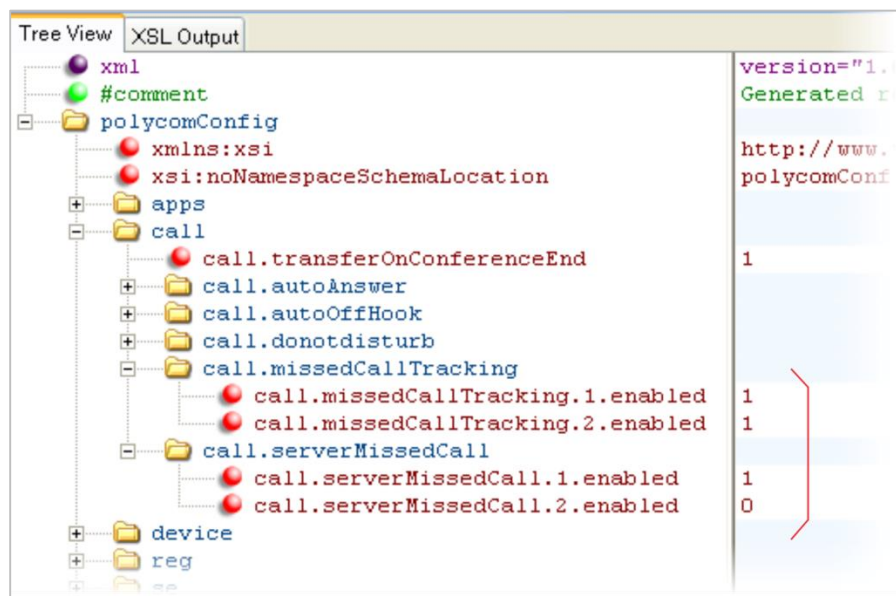
Table 8-9: Disabling Missed Call Notification

Parameter	Permitted Values	Default
call.missedCallTracking.x.enabled¹	0 or 1	1
<p>If set to 1, missed call tracking is enabled.</p> <p>If call.missedCallTracking.x.enabled is set to 0, then missedCall counter is not updated regardless of what call.serverMissedCalls.x.enabled is set to (and regardless of how the server is configured). There is no Missed Call List provided under Menu> Features of the handset.</p> <p>If call.missedCallTracking.x.enabled is set to 1 and call.serverMissedCalls.x.enabled is set to 0, then the number of missedCall counter is incremented regardless of how the server is configured.</p> <p>If call.missedCallTracking.x.enabled is set to 1 and call.serverMissedCalls.x.enabled is set to 1, then the handling of missedCalls depends on how the server is configured.</p>		

¹ Change causes handset to restart or reboot.

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.



Call Transfer

Two types of call transfer behavior is available on the 84-Series handsets. When Party A is talking to Party B and wishes to transfer the call to Party C:

- **Blind Transfer** Party A calls Party C and presses the Transfer softkey while the call is ringing. The call between Party A and Party B is ended when the Transfer softkey is pressed.
- **Consultative Transfer** Party A calls Party C while Party B is on hold. The call between Party A and Party B is ended when Party A presses the Transfer softkey.

By default, the Transfer softkey uses the consultative transfer functionality and the blind transfer is available on the Features softkey menu. This parameter allows you to change the Transfer softkey to use the blind transfer functionality.

Table 8-10: Using Call Transfer

Parameter	Permitted Values	Default
call.transfer.blindPreferred	0 to 1	0
If set to 1, the default softkey will be for blind transfer. If set to 0, the default softkey will be for consultative transfer. The other method is always available on the Features softkey menu when the handset is in call.		

Example Call Transfer Configuration

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



Call Lists

Table 8-11: Call List (Call Log) Parameters

Parameter	Permitted Values	Default
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.		
callLists.grouping	Unified, InOut, InOutMissed	Unified
Used by callLists.collapseDuplicates and callLists.size .		
Unified - apply the limit in callLists.size to the total number of call list entries, regardless of their type.		
InOut - apply the limit to incoming (Received + Missed) calls, and apply it separately to outgoing (Placed) calls. Thus, the total number of call list entries is actually twice that specified in callLists.size .		
InOutMissed - apply the limit to Missed calls, and then separately to Placed Calls, and then separately again to Received calls. Thus, the total number of allowed call list entries is three times that specified in callLists.size .		
callLists.collapseDuplicates	0 or 1	1
If 0, all calls are archived and presented in the call lists. If 1, consecutive incomplete calls 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 setting 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 handset's menu system. If you want to prevent the records from uploading to the provisioning server, enter a false URL in the <code>CALL_LISTS_DIRECTORY</code> 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.		

¹ Change causes handset to restart or reboot.

Miscellaneous Call Handling Parameters

The handset supports an optional per-registration feature that enables automatic call placement when the handset is off-hook.

The handset 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. For the Spectralink 84-Series handsets, x=6.

This per-site and per-handset configuration parameters are defined as follows:

Table 8-12: Call Parameters

Parameter	Permitted Values	Default
feature.ringDownload.enabled¹	0 or 1	1
If 0, the handset will not download ringtones when it starts up. If 1, the handset will download ringtones when it starts up.		
feature.nonVolatileRingerVolume.enabled	0 or 1	1
If 0, user changes to the ringer volume are reset to default when the handset reboots. If 1, user changes to the ringer volume are saved and maintained when the handset reboots.		
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.	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.dialtoneTimeOut¹	positive integer	60
The time is seconds that a dial tone will play before a call is dropped. If set to 0, the call is not dropped.		
call.enableOnNotRegistered¹	0 or 1	1
If 1, users can make calls when the handset is not registered. If 0, calls are not permitted without registration.		
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 overrule this feature if 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.stickyAutoLineSeize¹	0 or 1	0
If set to 1, the handset uses <i>sticky</i> line seize behavior. This will help with features that need a second call object to work with. The handset 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. 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. <i>Note:</i> This may fail due to glare issues in which case the handset may select a different available line for the call.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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.suppressFullUriDisplay.enabled¹	0 or 1	0
<p>This parameter affects the display on the connected in-call screen.</p> <p>If 0, the handset displays the full SIP URL of an incoming call when the call is from a server the phone is not registered to.</p> <p>If 1, the handset displays only the number of an incoming call when the call is from a server the phone is not registered to.</p>		
call.suppressIgnoreSoftkey¹	0 or 1	0
<p>If this is 1, the "Ignore" softkey will not show up on the incoming call screen. Instead, the softkey will be blank.</p>		

¹ Change causes handset to restart or reboot.

CMS 2.0

Configuration Management Software is a UI administrative tool to accelerate configuration of individual handsets and facilitate management of all handsets in a facility. It is a licensed product accessible through an account key which must be present on every handset.

Two parameters ensure that the handset's heartbeat can reach CMS and that CMS recognizes the handset. Both of these parameters must set to a non-Null value for the CMS heartbeat to be enabled.

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
cms.heartbeat.URL	A string of 0-256 characters	Null
<p>This is the URL to which the phone sends the heartbeats. It must be HTTPS.</p>		
cms.heartbeat.accountKey	A string of 0-100 characters	Null
<p>The unique id that identifies the customer. (Provided by Spectralink)</p>		

Conference Calls

Local conferences require a host handset, which processes the audio of all parties. All handsets support three-party local conferencing.

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
call.localConferenceEnabled¹	0 or 1	0
<p>If set to 0, the Conference and Join softkeys do not display during an active call and you cannot establish conferences on the handset.</p> <p>If set to 1, the Conference and Join softkeys display during an active call and you can establish conferences on the handset.</p>		

Parameter	Permitted Values	Default
call.transferOnConferenceEnd¹	0 or 1	1
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.		
call.localConferenceCallHold¹	0 or 1	1
If set to 0, a hold will happen for all legs when conference is put on hold. If set to 1, only the host is out of the conference, all other parties in conference continue to talk.		

¹ Change causes handset to restart or reboot.

Corporate Directory

You can connect your handset to a corporate directory server that supports the Lightweight Directory Access Protocol (LDAP) version 3. The corporate directory is a flexible feature and provides you with the parameters you can configure. Once set up on the handsets, the corporate directory can be browsed or searched. The user can call numbers and save entries retrieved from the LDAP server to the local contact directory on the handset.

Spectralink 84-Series handsets 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

Spectralink handsets support corporate directories that support server-side sorting and those that do not. For handsets that do not support server-side sorting, sorting is performed on the handset.



Tip: Better Performance With Server-Side Sorting

Spectralink 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 CS-14-19 *Corporate Directory Best Practices*.

A portion of the corporate directory is stored in flash memory on the handset. Spectralink 84-Series handsets have 256Mb of flash memory.

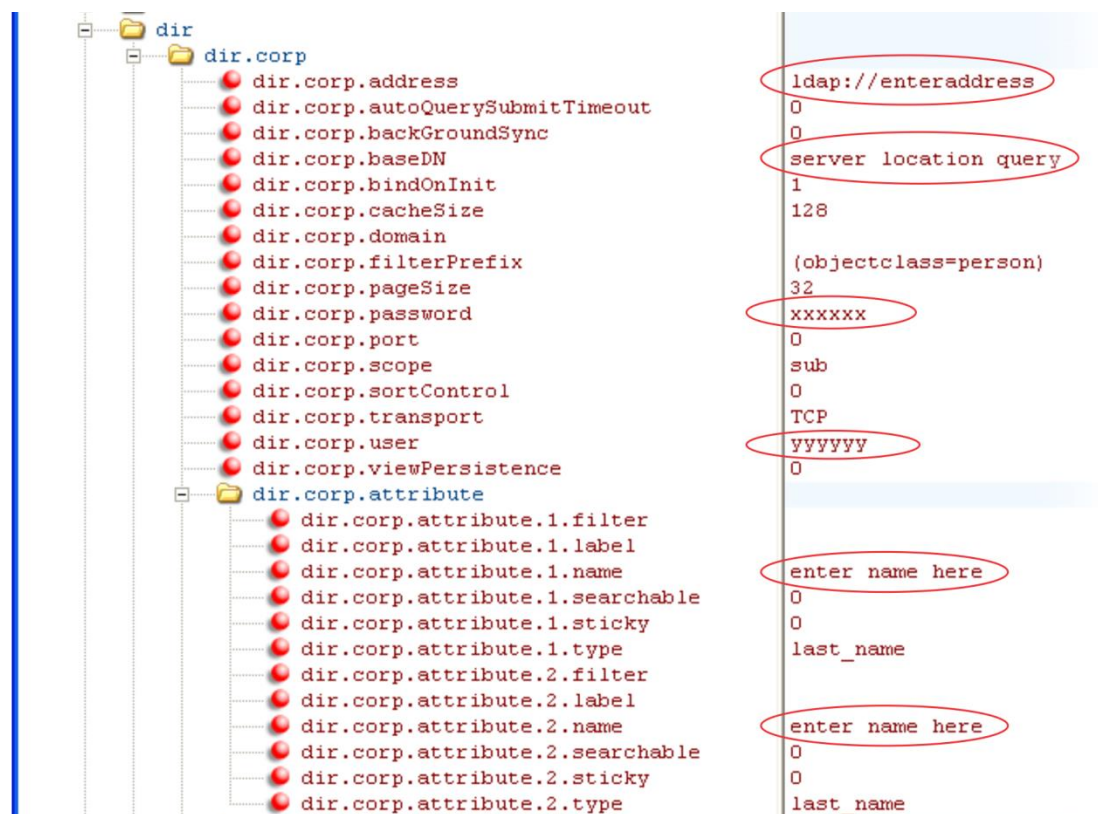
Table 8-13: Using the Corporate Directory

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
feature.corporateDirectory.enabled	0 or 1	0
If 0, the corporate directory feature is disabled. If 1, the feature is enabled.		
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 handset.		
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 handset. 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 handset, <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.		
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 in 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 handset.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
dir.corp.filterPrefix¹	UTF-8 encoded string	(objectclass=person)
The 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, 1 to 65535	389 (TCP) 636 (TLS)
The port that connects to the server if a full URL is not provided. When the value is set to 0, the default value will be used.		
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
Controls 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
Specifies whether a TCP or TLS connection is made with the server, if a full URL is not provided.		
dir.corp.useContainsSearch¹	0 or 1	0
When set to 0, the search will compare the entered characters against the starting characters of each field searched. This is the default behavior. When set to 1, the search will look for the entered characters anywhere inside each corporate directory field.		
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.		
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, givenName, telephoneNumber</code> .		

¹ Change causes handset to restart or reboot.

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.



Default Ring Tones and Alert Tones

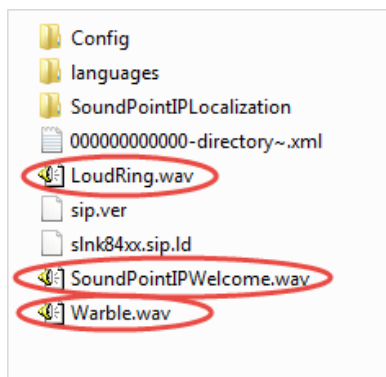
In different countries, different ring and tone patterns are used for a busy signal or reorder tone, for example. This section explains how to change the default ring tones and alert tones to the customary tones used in your area. See the sound effects pattern types table for more information.

Additionally, the handset can use 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. These are stored in volatile memory so the files will need to remain accessible should the handset 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

Your custom sampled audio files must be available at the path or URL specified by `saf.x` so the handset can download them at bootup. Include the name of the file and the **.wav** extension in the path. In the following example, you can see that the tones are at the top level of the zip file structure when the code is downloaded.



Summary

Parameter	Used to:
saf.x	Specify a path or URL for the handset to download a custom audio file
se.pat.*	Specify the name, type, and value for a custom sound effect

Table 8-14: 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. LoudRing.wav	
<p>Where “x” is the saf # that corresponds to the ringtone slot number you want the.wav file to appear in on the phone’s menu.</p> <p>The welcome tone is programmed to saf.1 which is menu slot 15. You can program this slot to a different welcome tone by programming saf.1. Slots 15 through 22 are available for custom tones. If Null, the handset will use a built-in file.</p> <p>If set to a path name, the handset will attempt to download this file at boot time from the provisioning server.</p> <p>If set to a URL, the handset 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>		

Specifying the saf.x parameter simply tells the phone to make the ringtone available in the ringtone menu list at the specified slot. Per the normal use of the saf parameter, the ‘x’ is an offset from the end of the default menu (slots 1-14). So saf.1 points to slot 15, etc...The table below provides the correspondences among saf#, menu slot# and pattern reference.

Notes on the Loudring.wav option

The LoudRing.wav file is a special case that makes this custom tone available on the phones rather than having to download a wav file to the phone from a provisioning server. Therefore, it is not necessary to have the file on the provisioning server since it is already on the phone.

Once this configuration option is loaded on the phone the LoudRing.wav will be available even if the phone cannot connect to the provisioning server.

Table 8-15: Default Sample Audio File Usage

SAF # Sampled Audio File Number	Menu slot #	Default Use (Pattern Reference)
1	15	Ringer 12 (se.pat.misc.welcome)
2	16	Ringer 13 (se.pat.ringer.ringer15)
3	17	Ringer 14 (se.pat.ringer.ringer16)
4	18	Ringer 15 (se.pat.ringer.ringer17)
5	19	Ringer 16 (se.pat.ringer.ringer18)
6	20	Ringer 17 (se.pat.ringer.ringer19)
7	21	Ringer 18 (se.pat.ringer.ringer20)
8	22	Ringer 19 (se.pat.ringer.ringer21)
9	23	Ringer 20 (se.pat.ringer.ringer22)
10	24	Ringer 21 (se.pat.ringer.ringer23)
11		Ringer 22 (se.pat.ringer.ringer24)
12 to 24		Not Used

The handset 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 8-16: Sound Effect Parameter

Parameter	Permitted Values	Default
se.appLocalEnabled¹	0 or 1	1
If set to 1, local user interface sound effects such as confirmation/error tones, will be enabled.		
se.stutterOnVoiceMail	0 or 1	1
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.		

¹ Change causes handset to restart or reboot.

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 8-17: Sound Effects Pattern Types

Instruction	Meaning
sampled (n)	Play sampled audio file <i>n</i>
Example: se.pat.misc.SAMPLED_1.inst.1.type = "sampled" (sampled audio file instruction type) se.pat.misc.SAMPLED_1.inst.1.value = "2" (specifies sampled audio file 2)	
chord (n, d)	Play chord set <i>n</i> (<i>d</i> is optional and allows the chord set ON duration to be overridden to <i>d</i> milliseconds)
Example:	

<i>Instruction</i>	<i>Meaning</i>
<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 d 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>	
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 8-18: 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 <code>callProg</code> , <code>ringer</code> , or <code>misc</code> .	
se.pat.cat.x.inst.y.type	sampled, chord, silence, branch
Type of sound effect, where <i>cat</i> is <code>callProg</code> , <code>ringer</code> , or <code>misc</code> .	
se.pat.cat.x.inst.y.value	String
The instruction: <code>sampled</code> – sampled audio file number, <code>chord</code> – type of sound effect, <code>silence</code> – silence duration in ms, <code>branch</code> – number of instructions to advance. <i>cat</i> is <code>callProg</code> , <code>ringer</code> , or <code>misc</code> .	

Call Progress Patterns

The following table shows the call progress pattern names and their descriptions:

Table 8-19: Call Progress Tone Pattern Names

<i>Call Progress Pattern Name</i>	<i>Description</i>
alerting	Alerting
bargeIn	Barge-in tone
busyTone	Busy tone
callWaiting	Call waiting tone
callWaitingLong	Call waiting tone long (distinctive)
confirmation	Confirmation tone

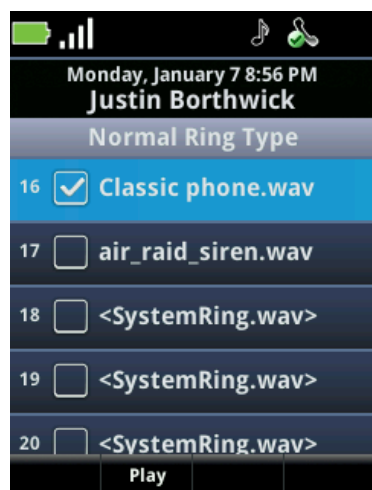
Call Progress Pattern Name	Description
dialTone	Dial tone
howler	Howler tone (off-hook warning)
intercom	Intercom announcement tone
msgWaiting	Message waiting tone
precedenceCallWaiting	Precedence call waiting tone
precedenceRingback	Precedence ringback tone
preemption	Preemption tone
precedence	Precedence tone
recWarning	Record warning
reorder	Reorder tone
ringback	Ringback tone
secondaryDialTone	Secondary dial tone
stutter	Stuttered dial tone

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 *Classic phone.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 *Classic phone* and is configured to use sampled audio file 12 (*Classic phone.wav*).



The following illustration shows the custom ring tone *Classic phone* as it displays on the handset menu:



Do Not Disturb

You can use the Do Not Disturb (DND) feature to temporarily stop incoming calls. Incoming calls received while DND is turned on are logged as missed. DND is enabled locally through the handset by navigating to **Settings> Feature Settings> Do Not Disturb**.

Table 8-20: Configuring Do Not Disturb

Parameter	Permitted Values	Default
call.rejectBusyOnDnd¹	0 or 1	1
If 1, and DND is turned on, the handset rejects incoming calls with a busy signal. If set to 0, and DND is turned on, the handset 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.donotdisturb.perReg¹	0 or 1	0
This parameter determines if the Do-Not-Disturb feature will apply to all registrations on the handset (globally), or apply on a per-registration basis. If 0, DND will apply to all registrations on the handset when it is active. If 1, the user can activate DND on a per-registration basis.		

¹ Change causes handset to restart or reboot.

Dual Tone Multi-Frequency (DTMF) Tones

The handset 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 handset 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.

Summary

<i>Parameter</i>	<i>Used to:</i>
<code>tone.dtmf.chassis.masking</code>	Specify if DTMF tones should be played through the speakerphone
<code>tone.dtmf.level</code>	Specify the frequency level of DTMF digits
<code>tone.dtmf.offTime</code>	Specify how long the handset should wait between DTMF digits
<code>tone.dtmf.onTime</code>	Specify how long the handset should play each DTMF tone for
<code>tone.dtmf.viaRtp</code>	Enable or disable DTMF encoding in an RTP stream

Table 8-21: DTMF Tone Parameters

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
<code>tone.dtmf.chassis.masking</code>¹	0 or 1	0
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.		
<code>tone.dtmf.level</code>¹	-33 to 3	-15
The level of the high frequency component of the DTMF digit measured in dBm0; the low frequency tone will be two dB lower.		
<code>tone.dtmf.offTime</code>¹	positive integer	50
When a sequence of DTMF tones is played out automatically, this is the length of time in milliseconds the handset will pause between digits. This is also the minimum inter-digit time when dialing manually.		
<code>tone.dtmf.onTime</code>¹	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).		
<code>tone.dtmf.viaRtp</code>¹	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 handset to restart or reboot.

DTMF Event RTP Payload

The handset 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 handset generates RFC 2833 (DTMF only) events but does not regenerate – or otherwise use – DTMF events received from the remote end of the call. Use the next table to set up this feature.

Table 8-22: Dual Tone Multi-Frequency (DTMF) Event RTP Payload

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
tone.dtmf.rfc2833Control¹	0 or 1	1
If set to 1, the handset will indicate a preference for encoding DTMF through RFC 2833 format in its Session Description Protocol (SDP) offers by showing support for the handset-event payload type. This does not affect SDP answers; these will always honor the DTMF format present in the offer since the handset has native support for RFC 2833.		
tone.dtmf.rfc2833Payload¹	96 to 127	127
The handset-event payload encoding in the dynamic range to be used in SDP offers.		

¹ Change causes handset to restart or reboot.

Emergency Calls

Emergency numbers can be programmed to appear on the Speed Dial list and Favorites menu. These numbers can be dialed when the handset is registered and available for calls. See [Call List](#) for more information.

When the phone's keypad is locked or a user is logged out, an emergency call can still be made if emergency dial numbers are programmed to appear on an Authorized Call menu.

Additionally, the PTT button on the 8453 handset can be programmed to be used as a duress button if PTT is not programmed. These two contingencies are explained below.

Emergency Dial via Authorized Call menu

When the phone is locked and requires a PIN before it will respond to keypad presses, a New Call softkey is available for emergency calls. When pressed, it opens the Authorized Call menu which lists numbers that can be reached while the phone is locked. Please see the [Phone Lock](#) section for information about phoneLock parameters that program the numbers to appear on the Authorized Call menu. For these numbers to go through, the phone must be registered and the numbers must be recognized by the call server. There are no phoneLock parameters that specify which server to use.

When User Profiles are deployed, when a user is logged out the phone is not available to make any calls except through the Authorized Call menu, accessed by pressing the Start button at the login screen. When users are logged out, the phone is not registered and calls usually cannot be made. However an emergency call server can be configured to allow anonymous calls and the numbers programmed to that server will go through. How to program the dialplan.routing.emergency parameters for server access and authorized numbers is covered in the User Profiles section [Placing Authorized \(Emergency\) Calls without Logging In](#).



Caution: What numbers are on the Authorized Call menu

All numbers programmed through `phoneLock.authorized.x` and `dialplan.routing.emergency.x` parameters will appear on the Authorized Call menu.

If the phone is registered, any number listed on the Authorized Call menu should go through. The `phoneLock` numbers will be routed through the SIP call server and the `dialPlan` numbers will be routed through the emergency call server.

If the phone is not registered, only those numbers programmed to the emergency call server that allows anonymous SIP calls will go through.

Emergency Dial via Duress Button

Emergency dial, also known as the “duress button” is enabled by programming the PTT key on the left side of the handset to function as a speed dial button programmed to call an emergency number. The Emergency Dial feature will allow the user to place the emergency call without having to unlock the keypad or unlock the handset. When the button is pressed twice within two seconds, a call is placed to the programmed number. A pop-up window on the display will inform the user that this mode is being activated.

If you enable Emergency Dial, PTT cannot be deployed. If both are enabled, Emergency Dial will overrule and PTT will not work.

The Emergency Dial feature places the call on Line 1. Therefore the PBX that line 1 is assigned to must be able to place a local emergency call. Any active call on line 1 will be preempted by the emergency call.

Two dial modes are configurable. Both modes are triggered by remapping the Talk key to a speed dial key. The software detects this remapping, which triggers the special handling of the key presses for this mode.

- The dial mode, which uses only a dial plan number for the Emergency Call, and
- the macro mode, which allows a more complex dial pattern for the Emergency Dial feature.

Dial Mode

This mode is configured in the `site.cfg` and `ptt.cfg` templates. A sample file is shown below:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- Example PTT Emergency Dial -->
<handsetConfig>
  <key
    key.20.function.prim="SpeedDial"
  />
  <ptt
    ptt.pttMode.enable="0"
```

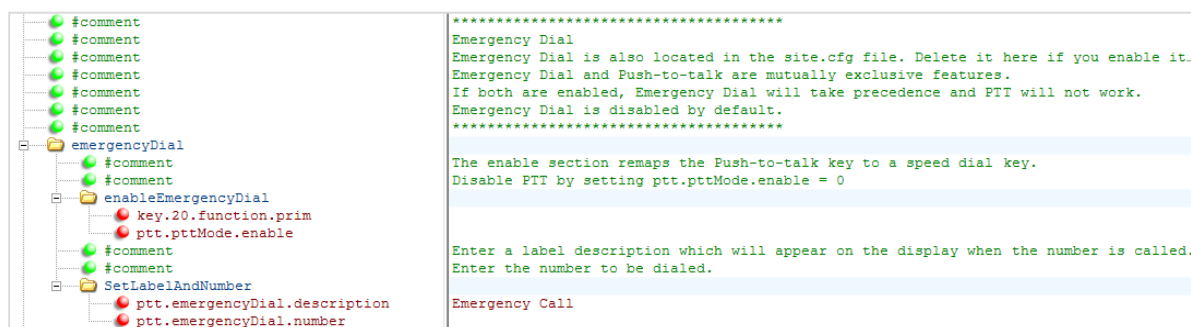
```

    ptt.emergencyDial.description="CODE BLUE Emergency"
    ptt.emergencyDial.number="6183"
  />
</handsetConfig>

```

The *key* section remaps the Talk key to a speed dial key. The *ptt* section turns off the *.pttMode*, provides a *.description* of the call which will appear in the To: line when the call is placed, and provides the extension *.number* to be dialed. You may customize the yellow highlighted fields for the display and the number to be dialed.

Dial Mode example in the ptt.cfg template (also available in the site.cfg template)



Macro Mode

The macro mode is configured by programming a macro as the dial string for the emergency number. EFK must be enabled. A sample file is shown below:

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
  <!-- Example PTT Emergency Dial Macro Mode-->
  <handsetConfig>
    <key
      key.20.function.prim="SpeedDial"
    />
    <ptt
      ptt.pttMode.enable="0"
      ptt.emergencyDial.description="CODE BLUE Emergency"
      ptt.emergencyDial.number="^6183$Tinvite$$Cwc$456$Cp2$$Tdtmf$"
    />
    <feature
      feature.enhancedFeatureKeys.enabled="1"
    />
  </handsetConfig>

```

The difference between the two files is in the *.number* parameter. This file uses a macro to describe the dialing sequence to be used. The example shown above will place a SIP call to 6183, wait for the call to be connected, wait for two seconds, and then send DTMF signaling for 456.

The first entry in the macro, **6183**, is the extension the call will be placed to. The entry, **\$Tinvite\$**, specifies that the call will be placed by using the SIP Invite method. **\$Cwc\$** specifies that the macro should pause until a connection is established with the far end. This could also be specified as **\$Cwaitconnect\$**. **456** specify the next digits to be dialed. **\$Cp2\$** or alternatively **\$Cpause2\$** causes a two second delay to occur prior to dialing after the connection is established. This can be any integer value from 1 to 10. **\$Tdtmf\$** specifies DTMF dialing for the second series of digits.

The Enhanced Feature Key (EFK) macro language is described elsewhere in this document.

NOTE: Currently, the *.description* parameter is ignored for macro based dialing. The To: line will appear as “To:EFK Dial”.

NOTE: For the macro dialing mode to function, the Enhanced Feature Keys mode must be enabled.

Default Values

If the PTT Emergency Dial feature is activated via the remapping of the Talk key to a Speed Dial key, the default values for the *.description* and *.number* parameters are “Emergency Call” and “911” respectively.

Table: Emergency Dial Parameters

Parameter	Permitted values	Default
key.20.function.prim	SpeedDial	Null
Changes key 20, the PTT button, to a speed dial function.		
ptt.pttMode.enable	0 or 1	0
If Emergency Dial is enabled, disable PTT by setting this value to 0.		
ptt.emergencyDial.description	string	Emergency Call
Enter a label description which will appear on the display when the number is called.		
ptt.emergencyDial.number	string	911
Enter the number to be dialed.		

Enhanced Feature Keys

Enhanced Feature Keys (EFK) enables you to customize the functions of line keys and softkeys. You can use EFK to assign frequently used functions to line keys and softkeys or to create menu shortcuts to frequently used handset settings.

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 softkeys. The EFK macro language was designed to follow current configuration file standards and to be extensible. The macros are case sensitive.

Different rules apply to configuring EFK for line keys and softkeys. Before using EFK, please become familiar with the macro language shown in this section.

Summary

<i>Parameter</i>	<i>Used to:</i>
reg.x.callsPerLineKey	Specify at least two calls per line key
feature.enhancedFeatureKeys.enabled	Enable or disable Enhanced Feature Keys
efk.efklist.x.*	Specify the EFK List parameters
efk.efkprompt.x.*	Specify the EFK Prompts

Because line keys and their functions are linked to fields in the contact directory file - 000000000000-directory.xml (global) or <MACaddress>-directory.xml (per handset),- 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 Local Contact Directory.

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), in the EFK logging module.
- 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 softkey 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).

Table 8-23: Enhanced Feature Key (EFK) List Parameters

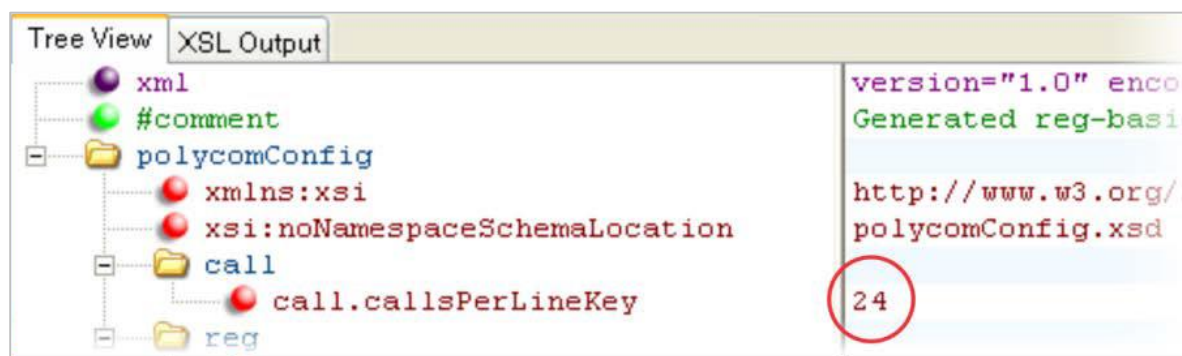
Parameter Name	Permitted Values	Default
efk.efklist.x.action.string	string	Null
The action string contains a macro definition of the action that the feature key will perform. If EFK is enabled, this parameter must have a value (it cannot be Null). For a list of macro definitions and example macro strings, see Understanding Macro Definitions.		
efk.efklist.x.label	string	Null
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.		
efk.efklist.x.mname	string	expanded_macro
The unique identifier used by the speed dial configuration to reference the enhanced feature key entry. Cannot start with a digit. <i>Note</i> that this parameter must have a value, it cannot be Null.		
efk.efklist.x.status	0 or 1	0
If 0, key x is disabled. If 1, the key is enabled.		

Table 8-24: Enhanced Feature Key (EFK) Parameters

Parameter Name	Permitted Values	Default
feature.enhancedFeatureKeys.enabled	0 or 1	0
If 0, the enhanced feature keys feature is disabled. If 1, the feature is enabled.		
efk.version	2 (1 for SIP 3.0 and earlier)	2
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.		

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. Failure to do so will prevent the macro from activating as additional calls per line key are used when activating a macro.



Find the enhanced feature keys feature in the everything.cfg template file. Use this file as the source for the EFK parameters. Copy/paste or drag them to the .cfg file you are developing for your facility.



In the following illustration, the EFK parameters are located in the everything.cfg 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 (1) 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 (1). The `label` on the user prompt has been defined as `Enter Number:` and this prompt will display on the handset 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.

dir	
efk	
efk.version	2
efk.efklist	
efk.efklist.1.label	Call Park
efk.efklist.1.mname	callpark
efk.efklist.1.status	1
efk.efklist.1.action.string	*681955
efk.efklist.2.label	Call Retrieve
efk.efklist.2.mname	callretrieve
efk.efklist.2.status	1
efk.efklist.2.action.string	*881955
efk.efkprompt	
efk.efkprompt.1.status	1
efk.efkprompt.1.label	Enter Number:
efk.efkprompt.1.userfeedback	visible
efk.efkprompt.1.type	numeric
efk.efkprompt.1.digitmatching	none
efk.efkprompt.2.status	1
efk.efkprompt.2.label	Enter Number:
efk.efkprompt.2.type	numeric
efk.efkprompt.2.userfeedback	visible
efk.efkprompt.2.digitmatching	none
feature	
key	
keypadLock	

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, i.e. as it is read from left to right. If necessary, user input is collected before any action is taken. The action string can contain the following fields.

To aid in the understanding of each of the following fields we will reference an example macro action string:

```
$LCallPark$20371$P1N4$$Tprefer$$Cwaitconnect$
```

Note that this example does not contain all actions.

Macro Actions and Descriptions

\$L<label>\$

This is the label that describes the macro action to be performed and will be displayed on the line key or softkey on the handset. The value can be any string including the null string (in this case, no label displays). This label will be used if no other label has been defined (up to the point where this field is parsed in the macro). 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.

In the above example, the label portion is the `$LCallPark$`. This field is not required but can be helpful to include when later referencing or troubleshooting macro setup.

digits

The digits to be sent. The appearance of this parameter depends on the action string. If no digits are to be sent it may not be necessary to include them. For example, if the macro includes a prompt to gather digits from the user then it may not be necessary to supply digits. Valid digits are 0-9 and '*' and '#'.
 In the above example, the digits being sent by the macro action string are 20371. Note that the digits do not get entered into the string with the '\$' on either end of the digits. This is because the digits are not a function that must be interpreted by the handset.

\$C<command>\$

This is a command that will be performed while processing the macro action string. It can appear anywhere in the action string and may appear multiple times. Supported commands (or shortcuts) include:

- hangup (hu)
- hold (h)
- waitconnect (wc)
- pause <number of seconds> (p <num sec>) where the maximum value is 10

Note that the shortcut version of each command can be entered into the macro string instead of the full command name.

In the above example, the command portion of the macro action string can be identified by the capital 'C'. In this case we are issuing a wait connect command, *\$Cwaitconnect\$*. The wait connect command causes the handset to stop processing the macro action string until the call is connected to the far end. Using it at the end of the line like this means that the handset will process everything before and then stop to wait for the call to connect. Since we were collecting digits from the user the handset will not send these digits until the call is connected.

\$T<type>\$

The type of action that will be performed if the macro action string is intended to initiate a call. Multiple actions can be defined. Supported action types include:

- invite
- dtmf
- refer

Note: Spectralink 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. The refer method is the equivalent of a call transfer. In the above example, the type of action being performed is, *\$Trefer\$*. Because this example is for a call park scenario the refer action type is used because we are going to transfer the active call on the handset to a network park location.

\$M<macro>\$

If you need to define multiple macros you can embed macros into a single macro string by using this option. The <macro> string must begin with a letter rather than a digit. If the macro name is not defined, the execution of the entire action string fails. Note that the <macro> is the actual name of another macro, not the macro action string. You would define this embedded macro in the same you are defining this current macro. The macro name is the efk.efklist.*.mname parameter.

\$P<prompt num>N<num digits>\$

You can use this option to cause the macro to prompt the user for additional input. That might be a name, a phone number or whatever you need them to enter. See Prompt Macro Substitution for more details on how to use this option.

In the above example, we defined a user prompt using, *\$P1N4\$*. This method uses the Prompt Macro Substitution configuration to create the text that is displayed on the handset to prompt the user for more input. It then collects the input, in this case, a maximum of 4 characters, and feeds that back into the macro for additional processing.

\$S<speed dial index>\$

Using this option you can define a directory entry as the location to collect the input needed to process the macro. Only digits are valid for the <speed dial index> as only digits are used in the directory to define an index location. The input collected is found in the `contact` field of the local directory entry pointed to by the index number.

\$F<internal function>\$

This particular option can be very powerful and equally useful when defining a macro. There are a number of internal handset functions that are pre-defined that you can call using this option. For example, you can use the internal function that allows you to navigate on the handset or adjust the handset volume.

URL

Entering a URL into a macro string can be used to cause the handset to browse to a specific web page or to access a server location using the handsets browser. Only one URL per action string is supported.

Prompt Macro Substitution

The `efk.efklist.x.action.string` can be configured to include a prompt using, **\$PnNn\$** where:

- **P** is the prompt **n** as defined by `efk.efkprompt.x` where `n=x`.
- **N**, where **n** 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** softkey to complete data entry.

The macros provide a generic and easy to manage way to define the prompt to be displayed to the user and the maximum number of characters that the user can input. The macros are case sensitive so you will need to ensure that the P and N are both capitalized.

If a macro attempts to use a prompt that is disabled, the macro execution fails. A prompt is not required for every macro.

Enhanced Feature Key (EFK) Prompt Parameters

Parameter Name	Permitted Values	Default
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 numeric, the characters are interpreted as numbers. If set to text, the characters are interpreted as letters. If Null, numeric 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 numeric and text is not supported.		
efk.efkprompt.x.userfeedback¹	visible or masked	visible
The user input feedback method. If set to visible, the text is visible. If set to masked, the text displays as asterisk characters (*), this can be used to mask password fields. If Null, visible is used. If this parameter has an invalid value, this prompt, and all parameters depending on this prompt, are invalid.		

¹ Change causes handset to restart or reboot.

Expanded Macros

Expanded macros are prefixed with the ^ character and are inserted directly into the local directory `contact` field. For more information, see 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.
- ^ This character indicates that the following characters represent the expanded macro (as in the action string).

Example Macro

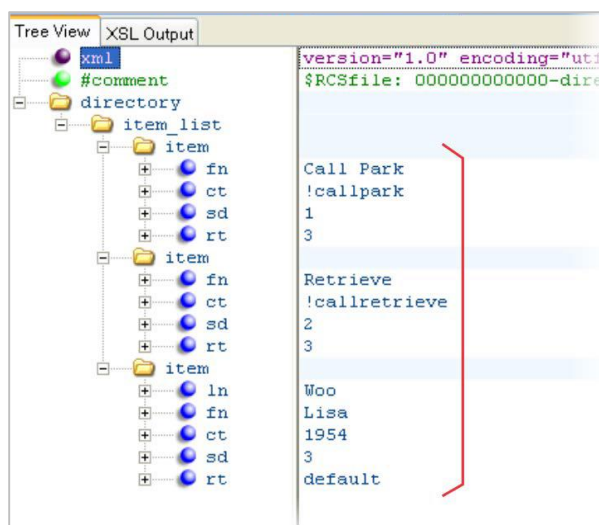
The action string:

```
$Changup$*444*$P1N4$$Tinvite$$Cwaitconnect$$P2N3$$Cpause2$$Tdtmf$$Changup$
```

is executed in order as follows:

- 1 The handset terminates any active call.
- 2 The handset receives digits *444*.
- 3 The user is prompted for 4 digits. For example, 1234.
- 4 The handset issues a new call to *444*1234 using the INVITE method.
- 5 The handset waits until the call connects before processing the macro further.
- 6 The user is prompted for 3 digits. For example, 567.
- 7 The handset pauses for 2 seconds
- 8 The handset sends the collected digits, 567, using DTMF tones.
- 9 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 [Local Contact Directory](#).

All line keys will be represented under the Features softkey fly-out menu on the handset.

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.

Features Softkey Menu Options Customization

On the Spectralink handsets, you can customize the flyout menu of the Features softkey to add options. This feature is typically used to access frequently used functions. As with EFK line keys, you assign functions to Features options using macros.

You can configure the custom option to display depending on the handset's menu level or call state. For example, you can make a Call Park option available when the handset is in an active call state.

Custom options 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 handset is ringing.
- **Dial tone** You can hear a dial tone.
- **Proceeding** (or outgoing proceeding) This state starts when the handset sends a request to the network. It stops when the call is connected.
- **Setup** This state starts when the user starts keying in a handset number. This state ends when the Proceeding state starts.
- **Hold** The call is put on hold locally.

New Features options 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

Up to 10 custom options can be configured. If more softkeys are configured than fit on the handset's screen, a **More** softkey displays. Press the **More** softkey to view the remaining softkeys.

This feature is part of Enhanced Feature Keys (EFK) and you must enable the enhanced feature keys parameter to configure softkeys.

Summary

Parameter	Used to:
feature.enhancedFeatureKeys.enabled	To turn Enhanced Feature Keys on (required)
softkey.x.action	Specify the macro for a line key or softkey function
softkey.x.enable	To enable a custom softkey
softkey.x.insert	Specify the position of the softkey on the handset screen

<i>Parameter</i>	<i>Used to:</i>
<code>softkey.x.label</code>	Specify the text to display on the softkey label
<code>softkey.x.precede</code>	To position the custom softkey before the default softkeys
<code>softkey.x.use.*</code>	Specify which call states the softkey will display in
<code>softkey.feature.*</code>	To display softkeys for various handset features, including default softkeys

Note that `feature.enhancedFeatureKeys.enabled` must be enabled (set to 1) to use the Configurable Softkey feature.

Table 8-25: Softkey Customization Parameters

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
<code>softkey.feature.basicCallManagement.redundant</code>	0 or 1	1
Control the display of the Hold , Transfer , and Conference softkeys. If set to 0 and the handset has hard keys mapped for Hold , Transfer , and Conference functions (all must be mapped), none of the softkeys are displayed. If set to 1, all of these softkeys are displayed.		
<code>softkey.feature.callers</code>	0 or 1	0
If 1, the Callers softkey displays on all platforms. If 0, the Callers softkey is disabled for all platforms. The default is 0.		
<code>softkey.feature.directories</code>	0 or 1	0
If 1, the Dir softkey displays on all platforms. If 0, the Dir softkey is disabled for all platforms. The default value is 0.		
<code>softkey.feature.endcall</code>	0 or 1	1
If 0, the End Call softkey is not displayed. If 1, the softkey is displayed.		
<code>softkey.feature.mystatus</code>	0 or 1	1
If 0, the MyStatus softkey is not displayed. If 1, the softkey is displayed (if <code>pres.idleSoftKeys</code> is set to 1). Only used with Lync.		
<code>softkey.feature.newcall</code>	0 or 1	1
If 0, the New Call softkey is not displayed when there is an alternative way to place a call. If 1, the New Call softkey is displayed.		
<code>softkey.feature.simplifiedSignIn</code>	0 or 1	0
If 0, the SignIn softkey is not displayed. If 1 and <code>voIpProt.server.x.specialInterop</code> is <code>lync2010</code> or <code>lync2013</code> , the SignIn softkey is displayed. The Lync Base Profile sets it to 1 by default and signin options are available in the Features softkey menu. See the Lync Interoperability Guides for more information.		
<code>softkey.x.action</code>	macro action string, 256 characters	Null
The action or function for custom softkey x. This value uses the same macro action string syntax as an Enhanced Feature Key. For a list of actions, see Macro Action .		
<code>softkey.x.enable</code>	0 or 1	0
If 0, the softkey x is disabled. If 1, the softkey is enabled.		
<code>softkey.x.insert</code>	0 to 10	0
The position on the handset screen for softkey x. For example, if the value is 3, the softkey 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 softkeys, the key will be positioned last, after the other softkeys.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
softkey.x.label	string	Null
The text displayed on the softkey label. If Null, the label is determined as follows:		
<ul style="list-style-type: none"> • If the softkey performs an Enhanced Feature Key macro action, the label of the macro will be used. • If the softkey calls a speed dial, the label of the speed dial contact will be used. • If the softkey performs chained actions, the label of the first action is used. • If the softkey 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, softkey x is positioned in the first empty space from the left. If 1, the softkey is displayed before (to the left of) the first default softkey.		
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	0 or 1	0
If 0, the softkey is not displayed when the handset is in the x state. If 1, the softkey is displayed when the handset is in the x state.		

¹ Change causes handset to restart or reboot.

Example Softkey Configurations

This section provides a few examples of available softkey configurations.



Web Info: Using Configurable Softkeys

For more examples, see Technical Bulletin 42250: *Using Enhanced Feature Keys and Configurable Softkeys on Spectralink handsets*.

To map a chained list of actions to a softkey:

- 1 Configure speed dial index 2 in the contact directory file with a handset address. For example, enter '2900' in the contact (ct) field.
- 2 In the contact directory, enter '12' 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 handset.

A softkey **ChainAct** displays. Press **ChainAct** to dial the handset number 2900.

To map the Do Not Disturb Enhanced Feature Key sequence to a softkey:

- 1 Update the configuration file as follows:

```
softkey.1.label = DND
softkey.1.action = $FDoNotDisturb$
softkey.1.use.idle = 1
```

- 2 Reboot the handset.

A **DND** softkey is displayed on the handset when it is in the idle state. When the **DND** softkey is pressed, the Do Not Disturb icon is displayed.

To map a Send-to-Voicemail Enhanced Feature Key sequence to a softkey:

- 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 handset.

When another party calls, the **ToVMail** softkey is displayed. When the user presses the **ToVMail** softkey, 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 softkey in the handset's idle state that navigates to a handset's administrator settings. The soft is inserted in softkey position 3, after the default softkeys. Note the macro action string:

```
$FMenu$$FDialpad3$$FDialpad2$$FDialpad4$$FDialpad5$$FDialpad6$$FSoftKey1$
```



<softkey/>

The configuration parameter is defined as follows (where x=1 to a maximum number of defined softkeys).

<efk/>

Use the following three tables to configure the Enhanced Feature Key feature on your handset.

Table 8-26: Enhanced Feature Key (EFK) Parameters

Parameter Name	Permitted Values	Default
feature.enhancedFeatureKeys.enabled	0 or 1	0
If 0, the enhanced feature keys feature is disabled. If 1, the feature is enabled.		
efk.version	2 (1 for SIP 3.0 and earlier)	2
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.		

Table 8-27: Enhanced Feature Key (EFK) List Parameters

Parameter Name	Permitted Values	Default
efk.efklist.x.action.string		
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 Macro Action .		
efk.efklist.x.label	string	Null
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.		
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. <i>Note:</i> 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 <code>efk.x.action.string</code> contains types, this parameter is ignored. If Null, the default of INVITE is used.		

Table 8-28: Enhanced Feature Key (EFK) Prompt Parameters

Parameter Name	Permitted Values	Default
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.		

<i>Parameter Name</i>	<i>Permitted Values</i>	<i>Default</i>
efk.efkprompt.x.type¹	numeric or text	text
The type of characters entered by the user. If set to <code>numeric</code> , the characters are interpreted as numbers. If set to <code>text</code> , the characters are interpreted as letters. If Null, <code>numeric</code> 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 <code>numeric</code> and <code>text</code> is not supported.		
efk.efkprompt.x.userfeedback¹	visible or masked	visible
The user input feedback method. If set to <code>visible</code> , the text is visible. If set to <code>masked</code> , the text displays as asterisk characters (*), this can be used to mask password fields. If Null, <code>visible</code> is used. If this parameter has an invalid value, this prompt, and all parameters depending on this prompt, are invalid.		

¹ Change causes handset to restart or reboot.

Handsfree Settings

Spectralink handsets have built-in speakerphones and support Bluetooth v2.1 headsets with Enhanced Data Rate (EDR) and Headset Profile (HSP v1.2). You can enable and disable each of these options.

Table 8-29: Audio Options for the Handset, Headset, and Speakerphone

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
up.headsetOnlyAlerting	0 or 1	0
If 1, only an auxiliary or Wi-Fi headset is used for alerting (such as incoming call alerting), when the headset is present.		
up.headset.phoneVolumeControl¹	disable, enable, auto	auto
When a headset is connected to the handset, the handset's behavior with respect to volume control events from certain headsets is different. enable – The handset responds to volume up/down events from the headset by displaying the volume widget in the handset's user interface and adjusting the handset's internal volume. disable – The handset shall ignore volume up/down events from the headset; pressing the headset's volume controls has no effect on the handset. auto – The handset shall automatically select which of the above two behaviors to apply, based upon the type and model version of headset that is attached.		
up.analogHeadsetOption	0, 1, or 3	0
The Electronic Hookswitch mode for the handset'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.		
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 handset can use a Bluetooth headset.		
feature.bluetooth.enabled	0 or 1	1
If 0, the Bluetooth headset feature is disabled. If 1, the feature is enabled.		

¹ Change causes handset to restart or reboot.

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 the next table.



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 8-30: Bluetooth Headset Support

<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 handset can use a Bluetooth headset.		
feature.bluetooth.enabled	0 or 1	1
If 0, the Bluetooth headset feature is disabled. If 1, the feature is enabled.		

Language Support

The handset language is selectable by the user by navigating to **Settings> Basic Settings> Preferences> Language**. The default language is English but a different language may be configured.

Supported languages are: Simplified Chinese, Traditional Chinese, Danish, Dutch, English, French, German, Italian, Japanese, Korean, Norwegian, Polish, Brazilian Portuguese, Russian, Slovenian, International Spanish, and Swedish.

Table 8-31: Setting the Phone Language

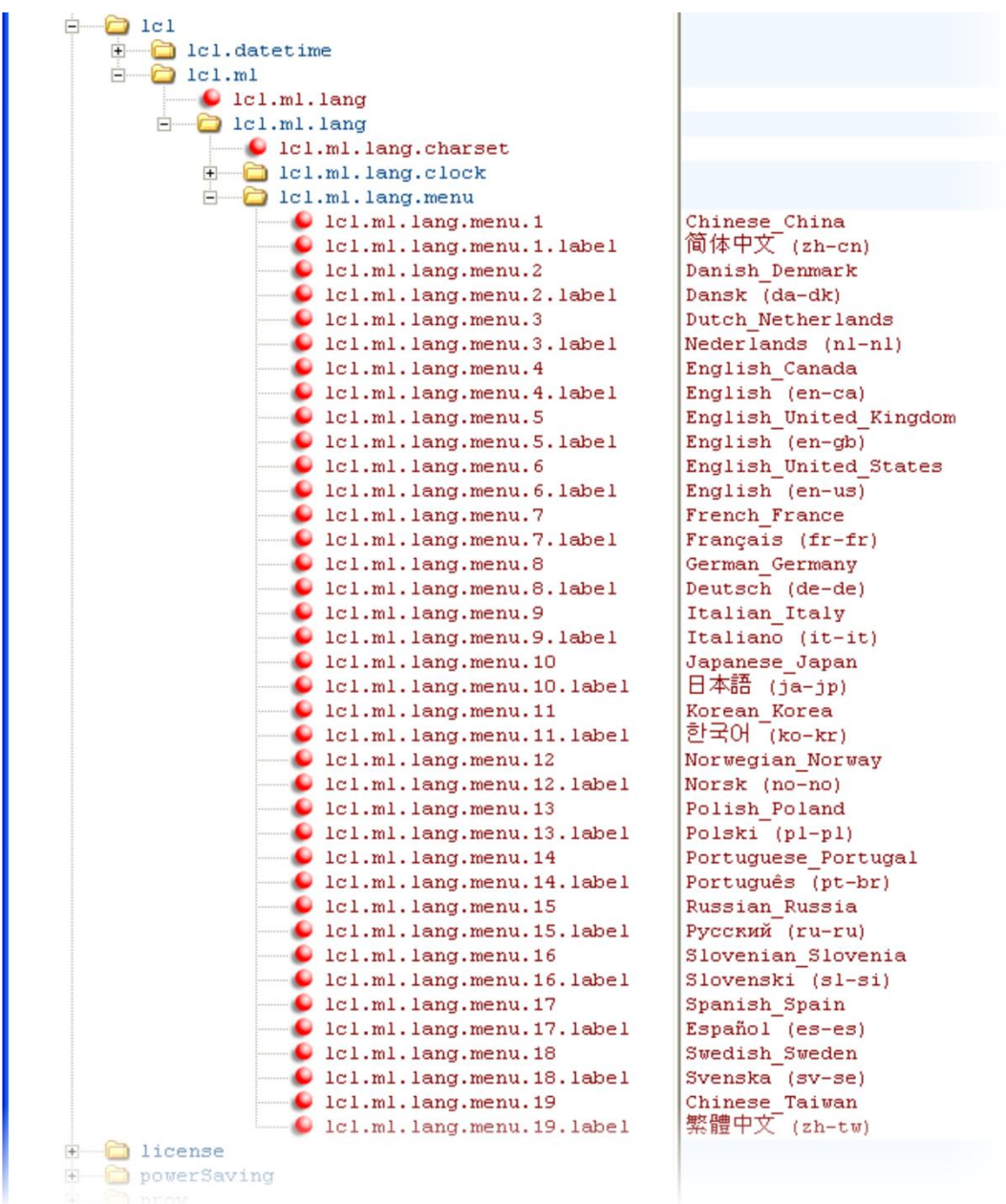
<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
lcl.ml.lang	Null or an exact match for one of the label names stored in lcl.ml.lang.menu.x.label	
If Null, the default internal language (US English) will be used, otherwise, the language to be used may be specified in the format of <code>lcl.ml.lang.menu.x.label</code> . Spectralink edits the following languages to match code updates: English_Canada English_United_Kingdom English_United_States French_France German_Germany		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
Italian_Italy Spanish_Spain		
lcl.ml.lang.charset¹	string	Null
The language character set.		
lcl.ml.lang.clock.x.24HourClock	0 or 1 or Null	Null
If parameter present, overrides lcl.datetime.time.24HourClock If 1, display time in 24-hour clock mode rather than am/pm. If Null, use value in lcl.datetime.time.24HourClock.		
lcl.ml.lang.clock.x.format	string which includes 'D', 'd' and 'M' and two optional commas	D,dM
If parameter present, overrides lcl.datetime.date.format; 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	[none]
If parameter present, overrides lcl.datetime.date.longFormat. If 1, display the day and month in long format (Friday/November), otherwise use abbreviations (Fri/Nov).		
lcl.ml.lang.list¹	a comma-separated list	
A list of the languages supported on the handsets.		

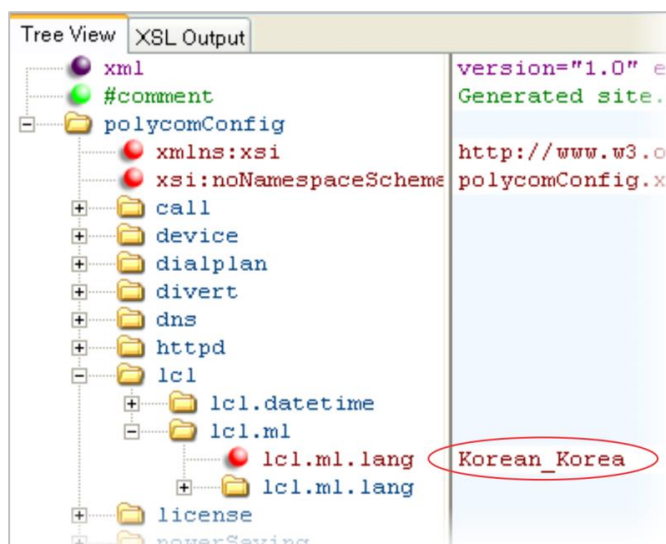
¹ Change causes handset to restart or reboot.

Example Phone Language Configuration

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



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



Once configured, the handset will use Korean characters. The language can be changed on the Settings menu.



Local Contact Directory

The Spectralink 84-Series handsets feature a contact directory you can use to store frequently used contacts. Open the Contact Directory by navigating to **Home> Contacts/Call Lists> Contact Directory**.

When it first starts up, the handset locates the contact directory by following a specific order:

- 1 First the handset looks in its local memory. If the user has populated the contact directory or makes any change to it, it will write to the MACaddress-directory.xml file on the central provisioning server and will use that file. If the user has not made any entries it will go to the second location.

If the contact directory has been populated, it will be written to the central provisioning server as an xml file named MACaddress-directory.xml every time any changes are made to it. See [Understanding the Files Written by the Handsets](#) for more information about how handset-written files work.

- 2 If there is no MACaddress-directory.xml file, the handset will look for a “seed directory” named 000000000000-directory.xml. This file can be provisioned at initial deployment to provide contact information to all handsets in the deployment. Once the handset finds this file, it populates the contact directory. If the user makes any changes to the contact information, the contact directory is written to the central provisioning server using its MACaddress as the filename as explained in step 1.

- 3 Failing the above 3 locations, the contact directory will remain empty.

The only time the handset “makes it” to #3 is when has been restored to factory defaults and there is no seed directory provisioned.

Provisioning the Seed Directory

The seed directory gives the administrator a way to populate the contact directory with global contact information at initial deployment. Unless a handset is returned to factory defaults, it will never again look for the seed directory because its local memory will be populated as in step 1 and any changes will be stored on the central provisioning server as in step 1.

Two features make the seed directory extremely useful in certain situations: you can provision an emergency number accessible on the speed dial list and you can apply distinctive incoming call treatment to specific contacts. You could also use it instead of a corporate directory and populate users’ contact directories with a full database of current personnel. (see [Speed Dial](#)) (see [Distinctive Incoming Call Treatment](#))



Contact directory is editable by the user

All entries in the users’ contact directories can be edited by the user unless you set the parameter to read-only.

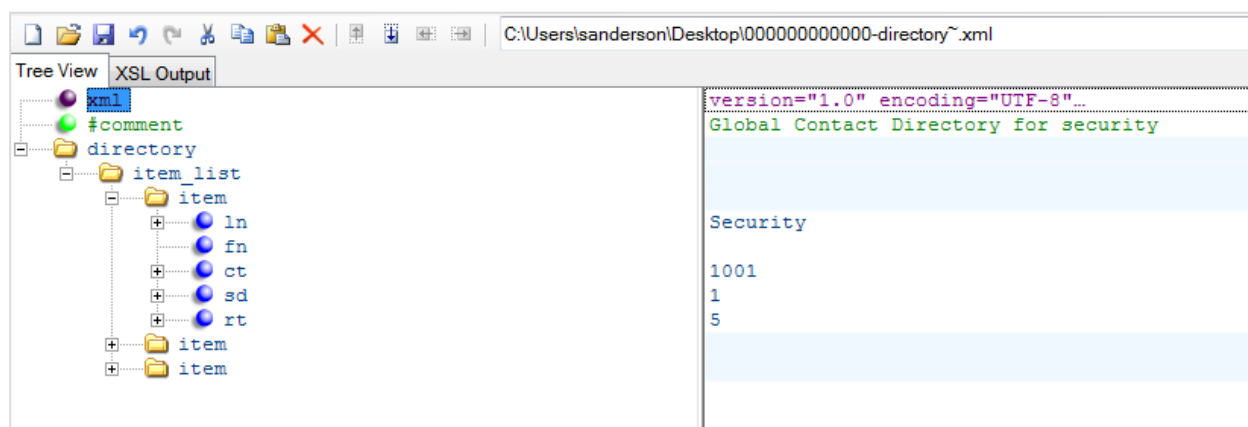
Setting up the seed directory:

- 1 Locate the seed directory template in the Config folder you downloaded with the software. It is named **000000000000-directory~.xml**.
- 2 Remove the tilde from the filename.
- 3 Populate the entries with your own contact information.

Table 8-32: Local Contact Directory elements

<i>Element</i>	<i>Definition</i>	<i>Permitted Values</i>
fn	First Name	UTF-8 encoded string of up to 40 bytes¹
The contact's first name.		
ln	Last Name	UTF-8 encoded string of up to 40 bytes¹
The contact's last name.		
ct ct2 ct3 ct4 ct5	Contact	UTF-8 encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL
Used by the handset 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. Up to 5 contact parameters may be configured, but only 1 is required (ct). Use dir.contact.attribute.x.label to configure the field name for each of the contact fields. E.g. cell phone, home phone, etc. Ensure the values entered for the contact align with the configured field names.		
sd	Speed Dial Index	Null, 1 to 9999
Associates a particular entry with a speed dial key for one-touch dialing or dialing from the speed dial menu.		
lb	Label	UTF-8 encoded string of up to 40 bytes¹
An element in the displayed name of the contact. The label field is usually used for a title. If the label field is not populated, the first and last names will display. . If the label field is populated, then the label will precede the first and last names: "label" "first name" "last name". E.g. Dr. John Smith will display when John Smith's label field is populated with Dr.		
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 overrules 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 overrules auto reject.		

Example seed directory with Security set to speed dial 1 and ring tone 5:



Configuring the Contact Directory

The defaults can be changed to accommodate specialized applications of the contact directory. You can enable or disable the local contact directory. If disabled it will not appear on the menu. You can specify read-only for the contact directory, in which case no contact information can be changed by the user. You can specify the maximum number of entries to conserve handset memory and you can specify whether the search of the directory is by last or first name.

The field names in the Contact Directory are fixed except for the contact field(s). The contact field value is the phone number for that entry. Up to five contact fields can be configured, for example a home, mobile and work number. These fields are intended to mimic corresponding fields that might be configured in the LDAP server. When Corporate Directory entries are saved, the values are saved to the corresponding fields in the Contact Directory. See the Tech Bulletin *Configuring Contact Fields*.

The field names for contact fields are drawn first from the `dir.contact.attribute.x.label` parameter, then from the `dir.corp.attribute` parameter and if these are not configured, the defaults are used, as shown below.

Table 8-33: Contact Directory elements

Parameter	Permitted Values	Default
feature.directory.enabled	0 or 1	1
If 0, the local contact directory is disabled. If 1, the directory is enabled.		
feature.urlDialing.enabled	0 or 1	1
You can enter a url as the contact number if the Mode softkey is enabled when editing the "Contact" field of a Contact Directory item. If 0, URL/name dialing is not available. If 1, URL/name dialing is available from private lines. Note: If enabled, unknown callers will be identified on the display by their phone's IP address.		
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>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
dir.local.contacts.maxNum¹	1 to 9999	9999
Maximum number of contacts allowed in the local contact directory.		
dir.search.field	0 or 1	0
If 0, search the contact directory by contact's last name. If 1, search by first name.		
dir.contact.attribute.x.label	String x characters	Primary, Contact 2, Contact 3, Contact 4, Contact 5
Sets the label for a contact field in the Contacts Directory. If x is 1, the default is Primary. The x value corresponds to the Contact number. E.g. if the x value or the label parameter is 2, the default value for the contact field is Contact 2. Note that the Primary contact number is used when setting a Speed Dial number to a contact or for default dialing. This parameter aligns with the ct values configured in the contact directory elements. I.e ct2 is the value for the Contact 2 field.		

¹ Change causes handset to restart or reboot.

Editing the Users' MACaddress-directory.xml File

In certain rare cases, an administrator might want to edit the user's contact directory file that has been written to the central provisioning server. Simply open the file with an xml editor and make the changes using the table above to identify the fields that need to be changed.

Specialized Caller Treatments

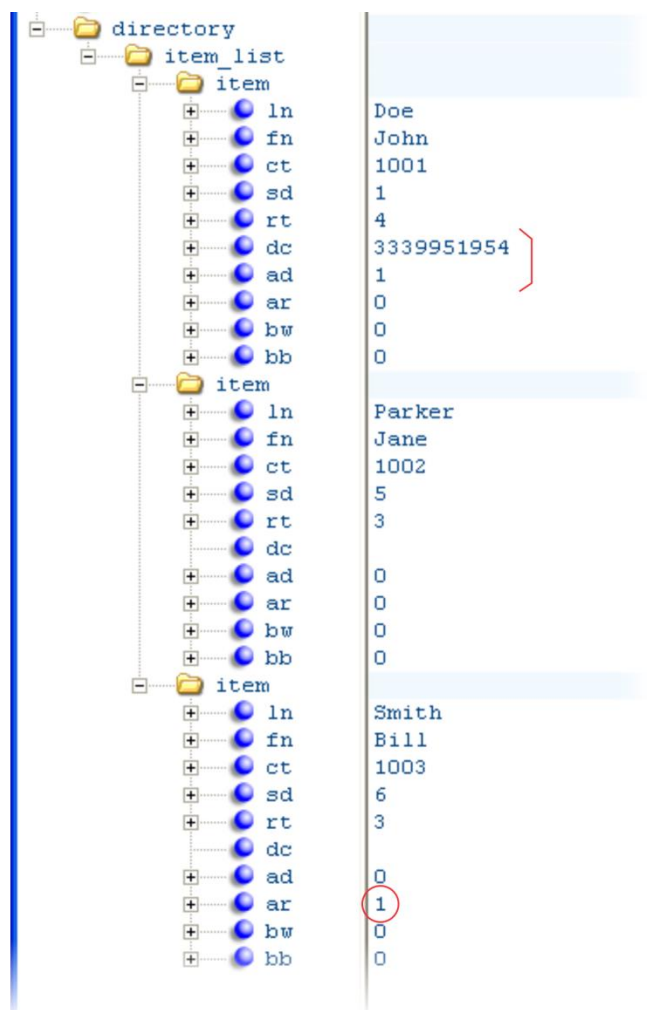
When setting up a seed directory or editing a user's contact directory, several specialized caller treatments can be applied. The user can also apply these treatments by opening and editing the contact in the contact directory. See [Table 8-32: Local Contact Directory elements](#).

Distinctive Incoming Call Treatment

You can apply distinctive treatment to specific calls and contacts in the contact directory. You can set up distinctive treatment for each of the contacts by specifying a **Divert Contact**, enabling **Auto-Reject**, or by enabling **Auto-Divert** for a specific contact.

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 overrules Auto Reject.

Speed Dial

You can link entries in the local contact directory to speed dial contacts on the handset. The range of speed dial numbers is from 1 to 9999. Usually this link is made by the user when the contact is set up or edited. However, it can be added in the seed directory or in the MACaddress-directory.xml files through the sd element as described in the Contact Directory Elements table.

Distinctive Call Waiting

You can use the alert-info values and class fields in the SIP header to map calls to distinct call-waiting types. You can apply three call waiting types: beep, ring, and silent. This feature requires call server support.

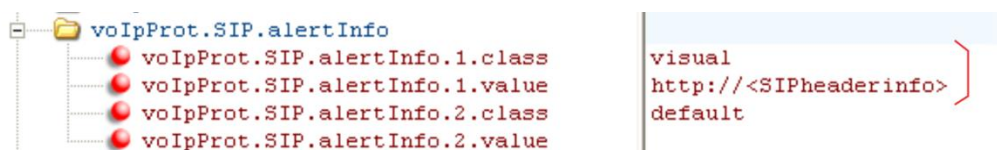
Table 8-34: Applying Distinctive Call Waiting

Parameter	Permitted Values	Default
voIpProt.SIP.alertInfo.x.class	See the list of ring classes in Ring Tones <rt/>	default

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.

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 handset to ring in a manner specified by `voIpProt.SIP.alertInfo.x.class`. In this example, the handset will display a visual LED notification, as specified by the value `visual`.



Location Services (Ekahau)

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). Location services are provided by the EPE 4.0 using Ekahau Location Protocol (ELP). For more information, see the Ekahau website description: [Ekahau Real Time Location System](#).

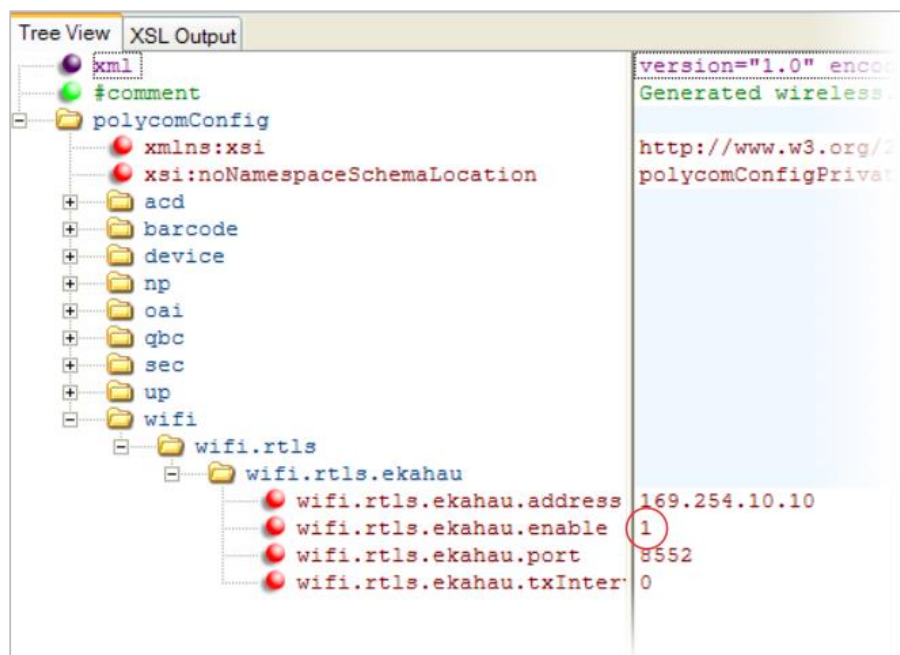
Table 8-35: Enabling Location Services

Parameter	Permitted Values	Default
wifi.rtls.ekahau.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 Ekahau Real-Time Location System (RTLS) is disabled. If 1, the Ekahau RTLS is enabled.		
wifi.rtls.ekahau.port	0 to 65535	8552
The port number of the Ekahau Positioning Engine.		
wifi.rtls.ekahau.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.		

Parameter	Permitted Values	Default
wifi.rtls.ekahau.txIntervalSeconds	10-600	Null
<p>Used in conjunction with Personal Alarms to control the interval with more precision. See the PersonalAlarms.cfg template and Personal Alarms configurable parameters. This setting will override any existing wifi.rtls.ekahau.txInterval parameter. If this setting is not defined or specified incorrectly, the existing wifi.rtls.ekahau.txInterval value will be used.</p> <p>During the alarm state, if RTLS is enabled, the handset will automatically use the shortest txInterval of 10 seconds regardless of the setting of any parameter. Once the alarm is cleared, the txInterval setting will revert to its former value.</p> <p>Note that this setting will overrule any interval set in the handset Administration Settings menu.</p>		

Example Location Service Integration Configuration

To use RTLS, enable the `wifi.rtls.ekahau.enable` parameter, as shown next. All other Ekahau parameter values shown in the following example are the default values.



Microsoft Exchange Calendar Integration

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 handset, configure the parameters in the next table.

You can access the feature from the **Applications** menu on the Spectralink handsets.

You will need valid Microsoft Windows credentials to access the Microsoft Exchange Calendar information on the handset. You can manage these credentials through the Login Credentials, which are available through **Home> Settings> Basic Settings> Login Credentials**.

You can view the calendar information in day or month format. The meeting details overlap the calendar view.

All possible handset 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 softkey.

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 and open the meeting details view. A tone will be played along with the reminder pop-up.



Web Info: Using Microsoft Exchange Calendar Integration

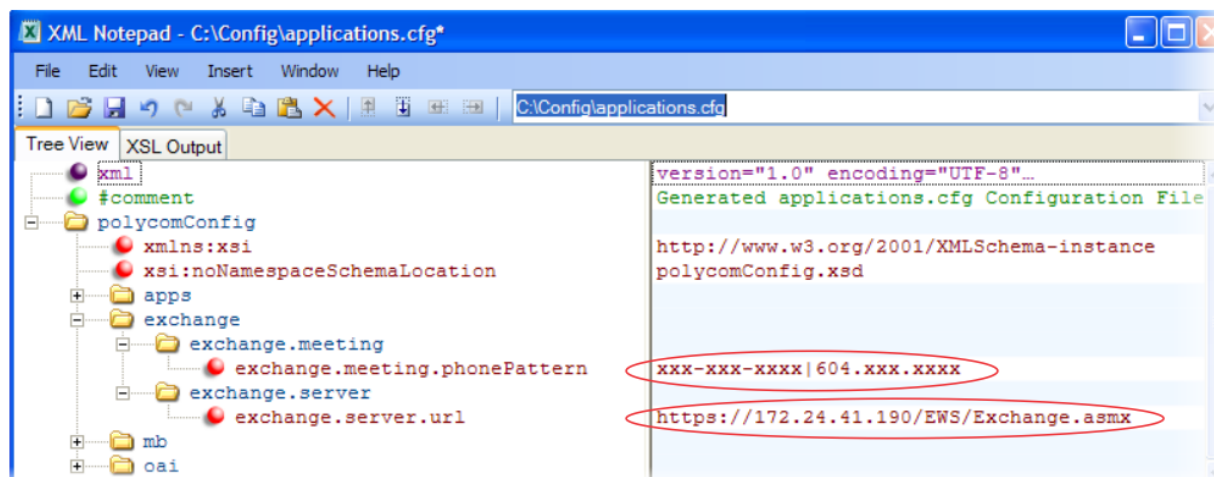
For user instructions on how to use calendar integration, refer to *the Spectralink 84-Series Wireless Handset User Guide*.

Table 8-36: Enabling Microsoft Exchange Calendar Integration

Parameter	Permitted Values	Default
feature.exchangeCalendar.enabled¹	0 or 1	0
If 0, the calendaring feature is disabled. If 1, the feature is enabled.		
exchange.server.url¹	String	Null
The Microsoft Exchange server address.		
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.		

¹ Change causes handset to restart or reboot.

After you enable the feature, specify the Microsoft Exchange Server address in **applications.cfg** template 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' handsets as a softkey. Then, participants can press the softkey to dial in to the meeting. You can specify multiple patterns, separated by a bar. In the following example, two patterns are specified.



Open Application Interface

Spectralink's Open Application Interface (OAI) enables you to use the Spectralink handsets to retrieve and respond to information on third-party computer applications.

Each handset that uses OAI features must be configured with its line number and MAC address in the OAI Gateway so that it can register to the OAI Gateway and receive messages from it. You can configure the OAI Gateway to recognize either the handset's MAC address or a "virtual MAC Address" substitute. The substitute is useful when deploying User Profiles where the handset user could use different handsets on different shifts. In this scenario the MAC address will not identify the user. Configuration of the oai.userid parameter allows you to configure an ID for a specific user. That way that user has the some ID whenever they are logged in.



Web Info: Using the Spectralink 8000 OAI Gateway

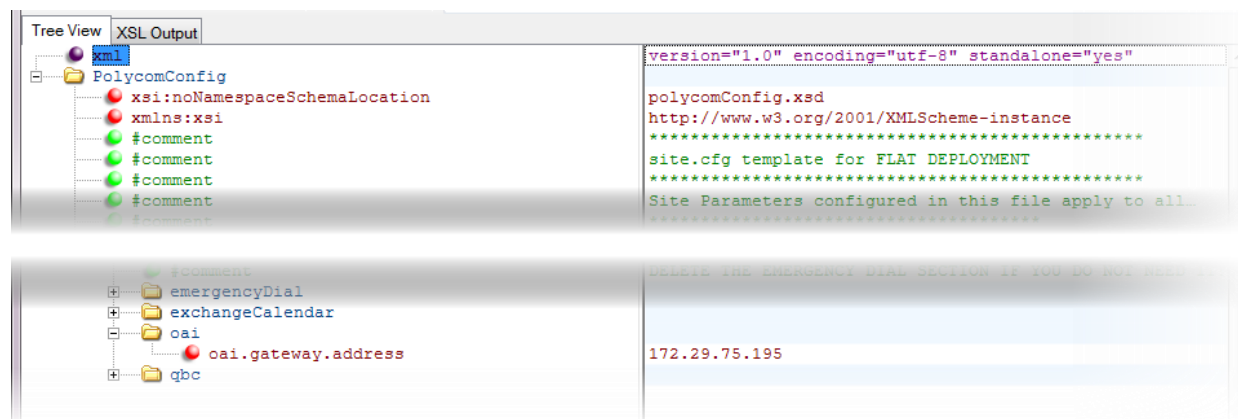
OAI v2.2 is supported by the Spectralink 84-Series handsets. For more information, see the *Spectralink 8000 Open Applications Interface (OAI) Gateway Administration Guide*.

Table 8-37: Configuring the Open Application Interface (OAI)

Parameter	Permitted values	Default
oai.gateway.address	IP address	Null
The address of the OAI server.		
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.		

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.



If deploying User Profiles, specify the User ID in the login.cfg file:



Passwords - User and Administrator

The parameters in this section regulate access to the admin menus on the handsets and user settings on the Web Configuration Utility. The handset will prompt you for a user or administrator password before you can access certain menu options. If the handset requires the administrator password, you may be able to use the user password, but you will be presented with limited menu options. If the handset 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. The admin password can be set in the wireless.cfg template when the handsets are first deployed. You may want to change the

user password for security reasons. Note that the user login password when deploying User Profiles is a separate password type. Please refer to the [User Profiles](#) section for more information about passwords and User Profiles.

Passwords are also used with Microsoft Lync servers, with certain security configurations and for access to configuration files using secure protocols. Please see the corresponding section for more information about provisioning handsets in secure environments.

Summary

<i>Parameter</i>	<i>Used to:</i>
sec.pwd.length.admin	Set the minimum length for the administrator password
sec.pwd.length.user	Set the minimum length for the user password
device.auth.localAdminPassword	Set the handset's local administrator password
device.auth.localUserPassword	Set the handset's local user password

Table 8-38: Local User and Administrator Password Settings

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
sec.pwd.length.admin¹	0-32	1
The minimum length for administrator passwords changed using the handset. Use 0 to allow null passwords.		
sec.pwd.length.user¹	0-32	2
The minimum length for user passwords changed using the handset. Use 0 to allow null passwords.		
device.auth.localAdminPassword	string (32 character max)	456
The handset's local administrative password. The minimum length is defined by sec.pwd.length.admin. If the value is Null, 456 will be used.		
device.auth.localUserPassword	string (32 character max)	Null
The handset user's local password. The minimum length is defined by sec.pwd.length.user. If the value is Null, 123 will be used.		

¹ Change causes handset to restart or reboot.

Table 8-39: Password obfuscation

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
webutility.outputKeysInClearText	0 or 1	0
<p>Note: Parameter only available in Release 4.5 and later.</p> <p>This parameter will cause passwords to be obfuscated when they are exported from the WebUI.</p> <p>0 = Any key or passphrase output in an exported configuration file or backup file created using the WebUI, will be output as "*****" if it has been set and as "" if not.</p> <p>1 = Any key or passphrase output in a exported configuration file or backup file created using the WebUI will be output in clear text if it has been set and as "" if it has not.</p> <p>NOTE: Passwords that are entered into menus on the handset are not shown at all. Only values that are entered into Config files are affected.</p>		

Affected parameters:

device.wifi.wep.key1	device.sec.configEncryption.key	qbc.connect.passphrase
device.wifi.wep.key2	device.logincred.password	sec.TLS.customDeviceKey
device.wifi.wep.key3	device.net.dot1x.password	tcplpApp.ice.password
device.wifi.wep.key4	device.pacfile.password	prov.login.defaultPassword
device.wifi.psk.key	device.prov.password	prov.login.localPassword
device.wifi.wpa2Ent.password	device.auth.localAdminPassword	apps.statePolling.password
	device.auth.localUserPassword	apps.push.password
		dir.corp.password
		diags.sshc.gateway.password

**Use the device.set parameter for each device.x parameter.**

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.

Personal Alarms

Workers can be at risk during security breaches or if personal incidents require immediate attention. Spectralink 8441 and 8453 handsets offer personal monitoring and duress call functionality, including “man down” alarms, “running” alarms and duress calls to an emergency number. Coupled with a security alarm application program, real time location information from the alarming handset can be displayed on security monitors and sent to other Spectralink 84-Series handsets for mobile response. The existing functionality of Location Services allows an alarming handset’s location to be pinpointed so that aid can be directed to the exact scene. When deployed in conjunction with a security alarm application, Spectralink Personal Alarms provide unparalleled support for isolated workers or other at-risk personnel in potentially threatening situations.

Duress call alarms can also be deployed within the functionality of the 8440 and 8452 models. The emergency dial feature can be programmed to sound a local alarm through the built-in speakerphone when an emergency call is dialed. Coupled with a security alarm application, this duress alarm can be used to identify the handset, the user and the location of the alarming handset.

**Warning**

The reliability of the Spectralink Personal Alarms application depends on the functionality and reliability of the greater infrastructure – the wireless LAN, the LAN, the call server, the central provisioning server, the server hosting location services, the central security system and its servers, the correct configuration of the Spectralink 84-Series handsets, correct installation and central provisioning server(s), and thorough training of personnel.

Four conditions of alerting can be activated and each is configurable to the requirements of the facility. If any of the running/tilt/still conditions occur, the handset will first warn the user about an impending alarm and if the user does not cancel the warning within a configurable number of seconds the handset will start to alarm.

	8440	8441	8452	8453
Duress	X	X	X	X
Running		X		X
Still		X		X
Tilt		X		X

- Duress – should the user press the emergency button on the left side of the handset a call is placed to an emergency number. A call can also be placed automatically if a running/tilt/still alarm is initiated,
- “Running” – the handset detects shaking in case a user runs for a configurable number of seconds,
- Tilt – the handset is tilted (not vertical) for a configurable number of seconds (aka “man down”),
- Still – the handset remains unmoved for a configurable number of seconds, potentially indicating the user is no longer moving.

Administrator Configurable Options

Alarm detection and sensitivity parameters are configurable only by an administrator through the configuration files on a central provisioning server. The `personalAlarm.cfg` file contains the parameters that are listed in this document. Spectralink Personal Alarms configuration options are not offered on the handset menus or through the Web Configuration Utility. They are only configurable through the central provisioning server configuration files. See the *Spectralink 84-Series Deployment Guide* for a full explanation of how to use a central provisioning server for Spectralink 84-Series handset provisioning and deployment.

Configurable options are

- The Motion events -- Running, Tilt, and Still -- may be configured for degree of sensitivity and duration before the warning becomes an alarm condition.
- The Duress alert is a call to an emergency number when the user either a) presses the Emergency Dial button on the left side of the handset twice within two seconds or b) presses and holds the button for x seconds. This button must be configured to dial an emergency number instead of its usual Push-to-talk function. The two features are mutually exclusive.
- The Motion events -- Running, Tilt, and Still -- can prompt an automatic emergency call.
- The emergency call can be configured to force use of the speakerphone instead of handset, headset or Bluetooth methods.
- A Suspend function can be configured to allow the user to temporarily disable the running/tilt/still sensing mechanism.

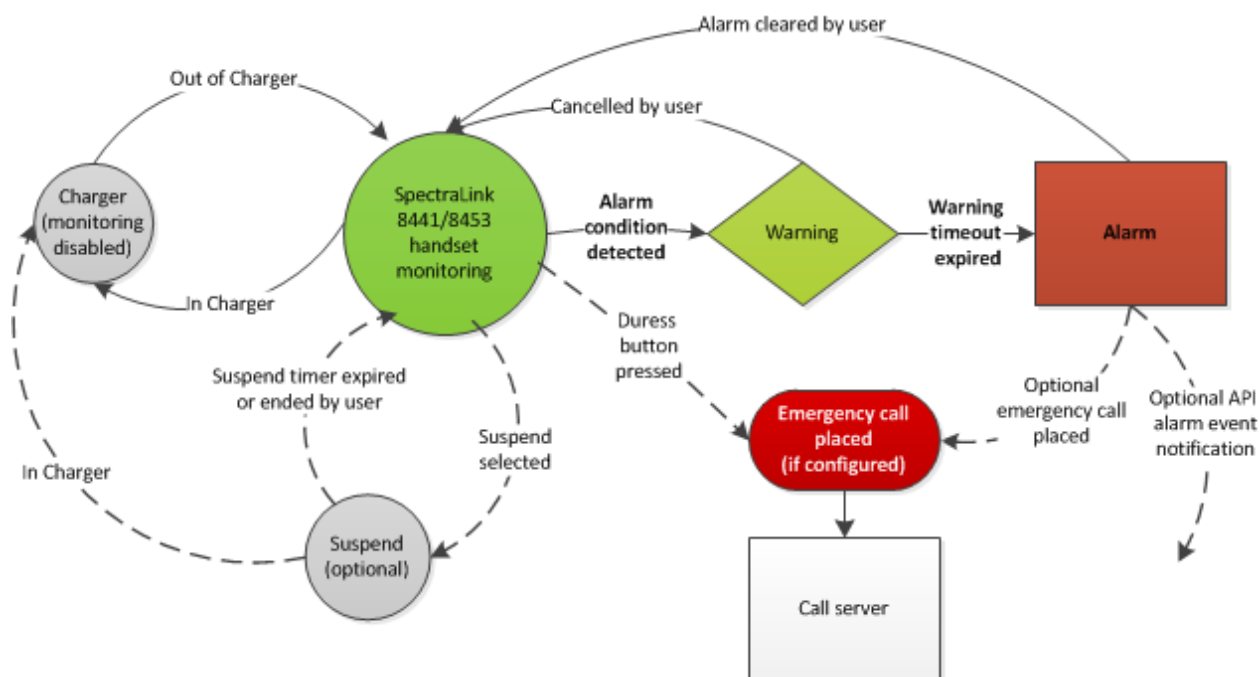
- A tone/vibrate/LED flash alarm notification can be configured for the handset when the warning/alarm event occurs.
- An XML API alarm event may be enabled/disabled for alarm events. See [Web Application Parameters](#).
- Already configured Location services are forced to the shortest txInterval of 10 seconds during an alarm condition.
- Tone/vibrate patterns for the Silent notification profile must be changed from the default of “silent” to a warning and alarm tone and/or vibration. Otherwise the alert and warning tones will not be played when the handset is in the Silent profile.
- The feature is disabled by ensuring the sensitivity for all motion events is set to zero.



Note: Emergency call functionality

The ability to make an emergency call can be configured on every Spectralink 84-Series handset model and is not unique to Spectralink Personal Alarms which are offered only on the 8441 and 8453 models. How to configure the Emergency Dial feature is described in detail in the *Spectralink 84-Series Deployment Guide*. Spectralink Personal Alarms can be configured to automatically generate an emergency call when a Running/Tilt/Still alarm is triggered if the Emergency Dial feature is also configured.

Diagram



User Experience

Optimal user experience requires an understanding of which Spectralink Personal Alarm features have been activated by the administrator. Users with 8441 and 8453 handsets should be advised if Running, Tilt, and/or Still alarms are enabled and all users should be advised if the Duress/Emergency Call functionality is enabled. All users need to know what the result will be if an alarm is activated, either intentionally or unintentionally. The user cannot permanently disable Spectralink Personal Alarms or turn it on and off.

The Spectralink Personal Alarms icon on the status bar indicates that the Spectralink Personal Alarms functionality is active. Colors and shapes signify its status:



= Spectralink Personal Alarms monitoring active



= Spectralink Personal Alarms alarm triggered



= Spectralink Personal Alarms temporarily suspended (if configured)

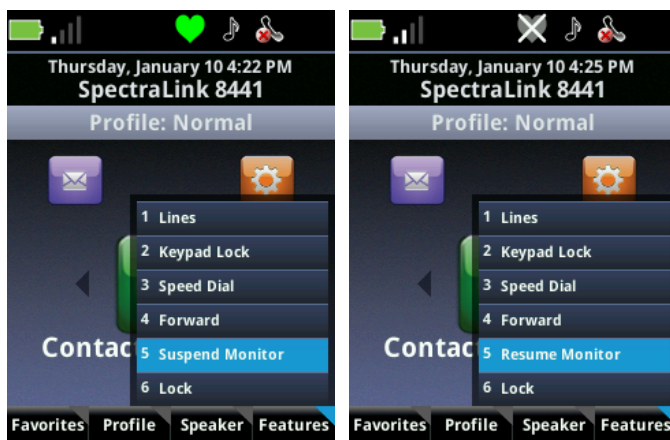
No icon = handset in charger (or the feature has been disabled by the administrator)



Note: Personal alarm icon for the 8440/8452 handsets

The Spectralink 8440/8452 models display the red icon during an Emergency Call until the alarm is cleared. No other icon is displayed.

- A security application may override a user or the notification profile setting.
- The administrator can program the warning and alarm alerts to sound even during the silent notification profile by overriding the default configuration for the silent profile. See the **np.xxx.alert** setting later in this document.
- The user can suspend Spectralink Personal Alarms running/tilt/still sensing mechanism for a period of time by selecting the **Suspend monitor** option on the Features softkey menu. The user may restart Spectralink Personal Alarms before the time elapses by selecting the **Resume monitor** option. Spectralink Personal Alarms suspend is a configurable option controlled by the administrator. The Duress button functionality is not suspended.



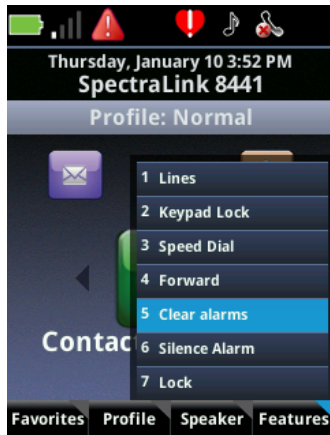
- If a running/tilt/still event has been detected, the handset goes into a “warning” state in which it presents a warning window and sounds the alert and/or vibrates. The user can cancel an impending alarm during the warning state by pressing the **Cancel** softkey and the alert audio will turn off and an alarm will not be sent to any configured security application and the handset resumes monitoring.



- If the user does not cancel during the running/tilt/still warning state, the handset goes into the alarm state. The alarm sounds and the alarm window appears. If configured, an XML alarm event is generated and sent to a security application, and an emergency call is placed.



- When you are in an alarm state, the alerting ring tone or vibration can be turned off by selecting the **Silence alarm** option on the Features softkey menu. Only the audio/vibration is stopped. The alarm state will continue until cleared. The flashing LED continues until the alarm state is cleared.
- Once an alarm state is reached, the user cannot retract the alarm but can cancel the alarming state and return the handset to the monitoring state by selecting the **Clear alarms** option on the Features softkey menu. Clear alarm returns the handset to the monitoring state and sends an xml notification of the end of the alarm. If an emergency call has been placed as well, the call must be terminated as would any call.



- If the Duress alarm is activated, a similar process occurs as for running/tilt/still alarms except there is no warning state and an emergency call is placed when the button is pressed twice within two seconds or with one long press. An emergency call preempts any existing active call. The emergency call is ended by pressing the END key or by the far end hanging up. The handset remains in the alarm state until manually cleared by the user.



- New alarm detection is inactive when the handset is connected to a charger – USB, Dual Charger or Speakerphone Dock. No icon displays during the inactive state. Once the handset is removed from the charger, monitoring begins and the icon turns green.
- Alarms must be manually cleared. They cannot be cleared by placing the handset into a charger.

Integration with Third Party Applications

Customers will receive the most benefit with Spectralink Personal Alarms by integrating the handsets with monitoring applications and real-time location-tracking (RTLS) systems. Monitoring applications, such as lone worker systems, can monitor workers in real time and implement sophisticated response management procedures such as notifying response teams in the case of an alarm. Such applications can be developed by third parties and customized according to a facility's needs.

Location tracking systems can provide pin-point accuracy in real-time to reduce the time spent finding the incident location. The following section describes several integration options when employing mandown handsets.

See the *Spectralink 84-Series Web Developer's Guide* for full information about API applications.



Note: Web applications parameters

See [Web Application Parameters](#) for information about applications parameters.

Implementation without application integration or Location Systems

Running/tilt/still alarms can be configured along with the automatic emergency call (aka Emergency Dial). In the event of any one of the three motion alarms being activated this will prompt an automatic emergency call to the configured emergency dial number. No application integration or RTLS is required for this first level of implementation. The emergency call provides notification of an alarm event. If the handset is configured for local notification, when an alarm is triggered the handset will emit noises to help rescuers locate the handsets.

Implementation with Location Systems

All Spectralink 84-Series handsets support integration with the Ekahau RTLS system. Handsets send periodic location information to an Ekahau server allowing the server to pinpoint the handsets' location. To maximize battery life, an administrator may set this update interval conservatively, e.g. 1 minute. In the event the handset enters an alarm state, the handset can be configured send updates more frequently, e.g. 10 seconds. This allows the Ekahau RTLS system to provide updated positions of the alarming handset more frequently. Additionally, if the Ekahau system is integrated into a management application these positions can be sent to responders.

Additional 3rd-party RTLS systems, e.g. Aeroscout, may also be able to provide the location of Spectralink 84-Series handsets.

Examples of implementation with third-party applications

Third-Party applications, such as Lone Worker Systems, will use the Spectralink 84-Series XML API to receive handset events and provide notifications to the user. Specific application capabilities are dependent on the developer and customer requirements. A typical application may monitor 84-Series handsets for Spectralink Personal Alarms events, provide alarm escalation logic to control who should receive notification, and then notify personnel providing relevant alarm information. Obviously the application sophistication and degree of integration with other systems may vary.

These examples assume that one or more of the Running, Tilt, and Still conditions are configured and that the Duress call is also configured (aka Emergency Dial). The automatic emergency call may or may not be configured.

- 1 When a handset enters the alarm state, if XML notification for alarms is configured, the handset will send an alarm event (`AlarmNotificationEvent`) to the application. A receiving application can use this to detect a handset has raised an alarm and perform the appropriate alarm response. The parameters set as part of the alarm event are covered in the XML API Detail section and the *Spectralink 84-Series Wireless Telephone Administration Guide* at the [Spectralink Support Website](#).
- 2 If configured, the emergency call triggered by the handset may be valuable identifying the severity of the situation and if the user of the alarming handset is responsive. And/or the application can use the XML API to send a message to the alarming handset to determine if the user intended to send an alarm and if the user is responsive or not.
- 3 If the application is integrated with an RTLS system, the application may attempt to locate the alarming handset before notifying the appropriate personnel to respond.
- 4 If the user of the alarming handset clears the alarm, if XML alarm notification is configured the handset will send an alarm notification to the application indicating the alarm has been cleared.
- 5 Assuming the application or application user needs to notify other personnel to respond to the alarm, the application can send appropriate messages to these response personnel. The XML API allows an application to deliver either simple messaging or

complex information such as a web-page with a site plan map showing the approximate location of the user.

- 6 To ensure a user is aware of these alarm response notifications the application can even override a user's volume settings.

Personal Alarms configurable parameters

Alarm detection and sensitivity parameters are configurable only by an administrator through the configuration files on a central provisioning server. The Spectralink Personal Alarms .cfg file contains the parameters that are listed in this document.

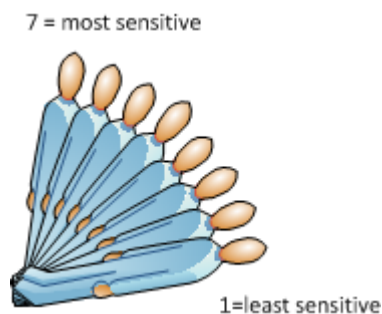


Caution

When a motion timeout setting is changed, the new value does not take effect until the handset detects a new occurrence of that motion condition and starts timing its duration to determine if warning should be raised. If the handset has already detected that motion condition (e.g. handset is currently tilted) and is currently timing the duration, the new timeout value will not be used until the handset exits that motion condition state (e.g. no longer tilted) and then enters it again.

Table 8-40: Personal alarms

Parameter	Permitted values	Default
up.PersonalAlarm.still.sensitivity 0 disable, 1 least sensitive, 7 most sensitive As the sensitivity increases the handset must be more stationary to trigger a no movement alarm. The specific sensitivity setting appropriate for a given application is site/user specific.	0-7	0
up.PersonalAlarm.still.timeout The still condition must persist for this amount of time before a warning occurs.	5-300 (secs)	7
up.PersonalAlarm.tilt.sensitivity 0 disable, 1 least sensitive, 7 most sensitive Indicates the degree of tilt from horizontal that triggers an alarm. This setting is modified by the timeout setting that determines the amount of time the position (or lower) is maintained before an alarm is triggered. In the most sensitive position setting, a slightly leaning position will trigger the alarm. In the least sensitive position setting, the body must be nearly horizontal before an alarm triggers. Experiment with these settings until you find the right sensitivity for your facility. Horizontal movement can interfere with the handset's ability to sense the tilt condition. If the tilt alarm is the only motion alarm configured, we recommend using a relatively high sensitivity setting.	0-7	0
up.PersonalAlarm.tilt.timeout The tilt condition must persist for this amount of time before a warning occurs.	5-300 (secs)	7



<i>Parameter</i>	<i>Permitted values</i>	<i>Default</i>
up.PersonalAlarm.running.sensitivity	0-7	0
0 disable, 1 least sensitive, 7 most sensitive As the sensitivity increases the handset requires less movement to trigger a running alarm. The specific sensitivity setting appropriate for a given application is site/user specific.		
up.PersonalAlarm.running.timeout	5-60 (secs)	7
The running condition must persist for this amount of time before a warning occurs.		
up.PersonalAlarm.suspendMonitoring	0- 300 (secs)	0
0, no suspension of monitoring is allowed, otherwise duration in seconds of the amount of time during which motion event monitoring is disabled.		
up.PersonalAlarm.warningTimeout	5-60 (secs)	10
This is the number of seconds a user has to disable the warning on the handset before the warning automatically triggers an alarm.		
up.PersonalAlarm.notificationEnable	0-1	1
Disable/enable local alarm notification (tone, vibrate, LED flash) on motion event detection in both warning and alarm states.		
up.PersonalAlarm.emergencyDialEnable	0-1	0
This parameter requires correct configuration of the Emergency Dial feature as detailed in the Spectralink 84-Series Administration Guide. This parameter is disabled by default. If enabled, an emergency call is automatically placed when a running/tilt/still alarm occurs. If disabled, and a running/tilt/still alarm occurs the emergency call will NOT be placed but the alarm state will be activated. Either way, the handset will transition into the alarm state and will generate an XML alarm notification if apps.telNotification.alarmEvent is enabled.		
up.PersonalAlarm.emerDialForceSpeakerPhone	0-1	1
This parameter requires correct configuration of the Emergency Dial feature as detailed in the Spectralink 84-Series Administration Guide. This parameter is enabled by default. Allows you to force the speakerphone on an emergency call. If 1 force speakerphone on emergency call, if 0 use normal audio termination routing rules (headset if connected, ...).		
ptt.emergencyDial.emergencyDialEnable	0-1	1
This parameter requires correct configuration of the Emergency Dial feature as detailed in Emergency . This parameter is enabled by default. If enabled, an emergency call will be placed if the duress button is pressed twice within two seconds. If disabled, and the duress button is pressed twice within two seconds, the emergency call will NOT be placed but the alarm state will be activated.		
ptt.emergencyDial.longKeyPressEnable	0-1	0
This parameter enables the long press method of activating the Emergency Dial function. It is disabled by default. When set (1), it overrides the default double press method (0).		
ptt.emergencyDial.longKeyPressDuration	1,2,3,4	2
If the long press method of activating the Emergency Dial function is enabled, this parameter sets the length of time in seconds that the key must be pressed in order for a call to be initiated.		
ptt.emergencyDial.notificationEnable	0-1	0
This parameter requires correct configuration of the Emergency Dial feature as detailed in Emergency . This parameter is disabled by default. Disable/enable local alarm notification (tone,vibrate,LED flash) on manual button press (a duress or emergency dial call). This parameter works in conjunction with Emergency Dial activation and applies to all handset models.		

Parameter	Permitted values	Default
wifi.rtls.ekahau.txIntervalSeconds	10-600	Null
<p>Used in conjunction with Location Services server to control the interval with more precision. See the RTLS.cfg template and Location Services (Ekahau). This setting will override any existing wifi.rtls.ekahau.txInterval parameter. If this setting is not defined or specified incorrectly, the existing wifi.rtls.ekahau.txInterval value will be used.</p> <p>During the alarm state, if RTLS is enabled, the handset will automatically use the shortest txInterval of 10 seconds regardless of the setting of any parameter. Once the alarm is cleared, the txInterval setting will revert to its former value.</p> <p>Note that this setting will overrule any interval set in the handset Administration Settings menu.</p>		
np.xxx.alert.PersonalAlarm.tonepattern	Any tone (see se.pat.misc)	Normal = misc2 silent = silent meeting = silent custom1 = custom2 0
np.xxx.alert.PersonalAlarm.vibration	0 or 1	
np.xxx.alert.PersonalWarning.tonepattern	Any tone (see se.pat.misc)	Normal = misc1 silent = silent meeting = silent custom1 = custom1 0
np.xxx.alert.PersonalWarning.vibration	0 or 1	
<p>If a consistent audible alarm or vibration is desired during the warning or alarm period, no matter the profile, the behavior must be set in the configuration file. These tones cannot be overridden by the user as these alerts do not appear on the editable alert menu for the notification profiles. This is especially important for the silent notification profile as the silent notification profile default behavior is silent for all alerts.</p> <p>Alert settings can be programmed in the personal alarm application to play a tone pattern or vibration according to the notification profile. See <i>Spectralink 84-Series Wireless Telephones Web Developer's Guide</i> for more information.</p>		

XML API Detail

An XML API allows you to use the telephone event notifications to develop a response application when the Spectralink Personal Alarms alarm is triggered.

The *Spectralink 84-Series Web Application Developers' Guide* explains how to install and use the Spectralink 84-Software Development Kit (SDK) to plan, create, and develop Web applications that will run on Spectralink 84-Series Wireless Telephones using Spectralink software 4.2.x or later. Consult that reference for more information.

Viewing an Alarm Event

The configuration parameter for this new event type is in the applications.cfg file.

Table 8-41: Alarm events

Parameter	Permitted Values	Default
apps.telNotification.alarmEvent	0 or 1	0
<p>If 0, alarm event notification is disabled. If 1, notification is enabled. When this parameter is enabled, an XML notification is sent when an alarm event occurs. Alarm events occur when Running, Tilt, and Still alarms go off and when Duress/Emergency Calls are made.</p>		

The Alarm event notifies an application that a (personal security) alarm condition has been detected or cleared on the handset.

Use the following format when viewing the alarm event:

```
<PolycomIPPhone>
<AlarmNotificationEvent>

  <PhoneIP>172.29.71.157</PhoneIP>
  <MACAddress>00907a0e4459</MACAddress>
  <BSSID>0023ebe4ebaf</BSSID>
  <StillAlarm>0</StillAlarm>
  <TiltAlarm>1</TiltAlarm>
  <RunningAlarm>0</RunningAlarm>
  <DuressAlarm>0</DuressAlarm>
  <TimeStamp>2012-12-10T08:11:25-07:00</TimeStamp>

</AlarmNotificationEvent>
</PolycomIPPhone>
```

Where

Phone IP is the IP address of the handset

MAC Address is the MAC address of the handset

BSSID is the MAC address of the AP the handset is currently using

StillAlarm is the current state of this alarm detector 0 = no alarm, 1 = alarm

TiltAlarm is the current state of this alarm detector 0 = no alarm, 1 = alarm

RunningAlarm is the current state of this alarm detector 0 = no alarm, 1 = alarm

DuressAlarm is the current state of this alarm detector 0 = no alarm, 1 = alarm

TimeStamp is the recorded time of the event

Configuration Template

The PersonalAlarm.cfg template is provided with the software download in the Config>Features folder. Use it to as a starting point to customize your deployment of this feature.

Tree View	XSL Output
<ul style="list-style-type: none"> xm:ns:xsi xsi:noNamespaceSchemaLocation #comment #comment #comment MotionAlarms <ul style="list-style-type: none"> up.PersonalAlarm.still.sensitivity up.PersonalAlarm.still.Timeout up.PersonalAlarm.tilt.sensitivity up.PersonalAlarm.tilt.Timeout up.PersonalAlarm.running.sensitivity up.PersonalAlarm.running.Timeout #comment #comment SuspendTimeout <ul style="list-style-type: none"> up.PersonalAlarm.suspendMonitoring up.PersonalAlarm.warningTimeout DuressButton <ul style="list-style-type: none"> up.PersonalAlarm.emergencyDialEnable Ptt.emergencyDial.notificationEnable up.PersonalAlarm.emerDialForceSpeakerPhone #comment #comment #comment LocationServices <ul style="list-style-type: none"> wifi.rtls.ekahau.txIntervalSeconds #comment #comment APIdetail <ul style="list-style-type: none"> apps.telNotification.alarmEvent #comment #comment #comment 	<pre> http://www.w3.org/2001/XMLSchema-instance polycomConfig.xsd ***** PersonalAlarm.cfg template ***** 7 5 7 5 7 5 5 Motion alarms are all disabled by default. One or more must be enabled in order for the Peraonal Alarm function to work. 300 60 1 1 1 You must enable the Emergency Dial Feature in order for these two parameters to work. See the ptt.cfg and site.cfg template for more info. 15 An RTLS.ekahau server must be deployed. See the RTLS.cfg template. 1 If configuring an API, include this parameter with the rest of the apps.x parameters required by the third party application. See everything.cfg in teh Troubleshooting folder for apps.x. </pre>

Phone Lock

The Phone Lock feature allows you to add an option to the Features softkey menu to lock the handset and require a password to unlock it. This can be important in certain settings where security is an important factor in handset deployment. Keypad lock is available for inadvertent key presses. This is a more advanced security feature.

Please review the **Enhanced Feature Keys** section for complete information about configuring additional menu keys. Enhanced Feature Keys must be activated to configure this feature.

When a handset is in a call, IM or OAI session and another call comes in, the handset behaves normally and only locks when it returns to the idle state for the configured amount of time.


A handset may be set to alert when it is locked or it may be set to DND. The following table details the different handset behaviors when Alert or DND is turned on.

Table 8-42: Phone Lock Behavior

	Alert	DND
Incoming Phone call	Call Alert screen with caller ID info. User presses green key and is prompted for user PIN, on successful entry call is answered. Phone automatically locks on end of call.	Voice call goes right to voice mail. Phone remains locked.
Incoming OAI	OAI alert screen (app can put more info on screen while ringing once ring ack'd). User presses green key and is prompted for user	No OAI alert screen displayed. Phone sends "user did not respond to ring" response immediately so next level escalation if

	<i>Alert</i>	<i>DND</i>
	PIN, on successful entry call is answered. Phone automatically locks on end of call.	appropriate can begin right away. Phone remains locked.
Incoming IM	IM alert screen with first IM message contents. User presses green key and is prompted for user PIN, on successful entry IM session is on hold. User presses resume and IM session proceeds. Phone automatically locks on end of call.	No IM alert screen displayed. Phone sends a "486" response which means "The user was contacted but is currently not willing or able to take the call." The first IM message is added to the conversation list and the missed message count is incremented so the user can see they missed a message. Phone remains locked.

Table 8-43: 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	Null
phoneLock.authorized.x.value The number or address for an authorized call recipient.	string	Null
<p>The numbers configured by these parameters appear on the Authorized Call menu when the New Call softkey is pressed while the phone is locked.</p> <p>Up to five (x=1 to 5) authorized contacts can be configured. Each contact requires a description that displays on the screen, and a phone number or address value for the handset to dial.</p> <div style="display: flex; align-items: center;">  <p>These calls will only go through if the phone is registered and the numbers are recognized by the call server.</p> </div> <p>Additional numbers configured in the dialplan may also appear on the Authorized Call menu. See Emergency Dial via Authorized Call menu for additional information.</p>		
phoneLock.browserEnabled	0 or 1	0
If 0, the browser or browser is not displayed while the handset is locked and the handset cannot receive any updates from a browser application. If 1, the browser is displayed while the handset is locked.		
phoneLock.dndWhenLocked	0 or 1	0
If 0, the handset can receive calls while it is locked. If 1, the handset enters Do-Not-Disturb mode while it is locked. The user can enable DND even if this parameter is set to 0 which will cause the handset to demonstrate the behavior described in the above table until the user disables DND.		
phoneLock.enabled¹	0 or 1	0
If 0, the handset lock feature is disabled. If 1, the handset lock feature is enabled. <i>Note:</i> To 'unlock' the handset 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 handset can be idle before it automatically locks. If 0, automatic locking is disabled.		
phoneLock.lockState	0 or 1	0
The value for this parameter indicates whether the handset is locked or unlocked and changes each time you lock or unlock the handset. If 0, the handset is unlocked. If 1, the handset is locked. Note that the handset 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.		

Parameter	Permitted Values	Default
phoneLock.powerUpUnlocked	0 or 1	0
Use this parameter to override <code>phoneLock.lockState</code> . If 0, the handset retains the value in <code>phoneLock.lockState</code> . If 1, you can restart, reboot, or power cycle the handset to override the value for <code>phoneLock.lockState</code> in the <code>MAC-phone.cfg</code> and start the handset in an unlocked state. You can then lock or unlock the handset locally. Spectralink recommends that you do not leave this parameter enabled as its use is primarily administrative.		

¹ Change causes handset to restart or reboot.

Provisional Polling of Spectralink Handsets

You can control how your handset handles automatic provisioning by configuring the parameters in [Table 8-44: Provisional Polling of Spectralink handsets](#).

You can set the handset's automatic provisioning behavior to be:

- **Absolute** The handset polls at the same time every day.
- **Relative** The handset polls every x seconds, where x is a number greater than 3600.
- **Random** The handset 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 handset starting up and the polling period. Afterwards, the handset will poll every x seconds.
 - If you set the polling period to be greater than one day, the handset polls on a random day based on the handset's MAC address.

For example:

- If `prov.polling.mode` is set to `rel` and `prov.polling.period` is set to `7200`, the handset polls every two hours.
- If `prov.polling.mode` is set to `abs` and `prov.polling.time` is set to `04:00`, the handset 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 handset polls randomly between 1am and 5am every day. This parameter is only used when `prov.polling.mode` is set to `random`.
- If `prov.polling.mode` is set to `abs` and `prov.polling.period` is set to `2328000`, the handset polls every 20 days.

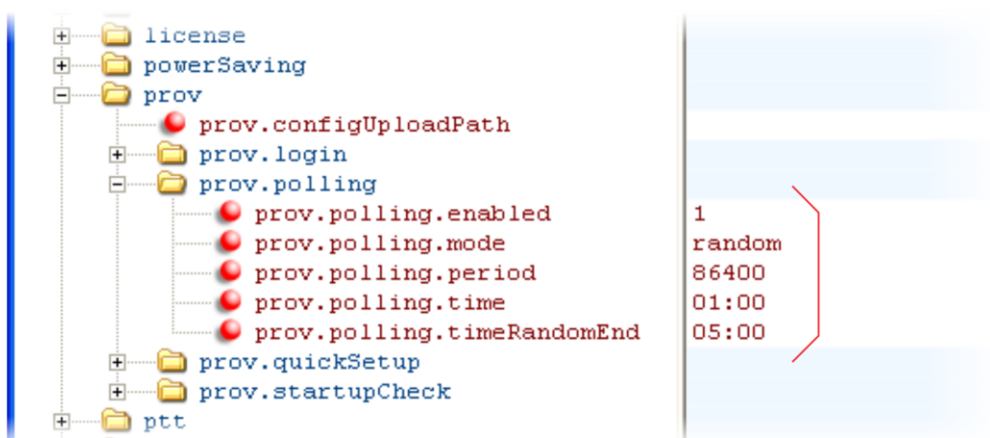
Table 8-44: Provisional Polling of Spectralink handsets

Parameter	Permitted Values	Default
prov.polling.enabled	0 or 1	1
If 0, the provisioning server is not automatically polled for upgrades. If 1, the provisioning server is polled.		

Parameter	Permitted Values	Default
prov.polling.mode	abs, rel, random	abs
<p>The polling mode.</p> <p>abs The handset polls every day at the time specified by <code>prov.polling.time</code>.</p> <p>rel The handset polls after the number of seconds specified by <code>prov.polling.period</code>.</p> <p>random The handset 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 handset's MAC address.</p>		
prov.polling.period	integer > 3600	86400
<p>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 handset 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 handset's MAC address.</p>		
prov.polling.time	hh:mm	03:00
<p>The polling start time. Used in absolute and random modes.</p>		
prov.polling.timeRandomEnd	hh:mm	Null
<p>The polling stop time. Only used in random mode.</p>		

Example Provisional Polling Configuration

The following illustration shows the default sample random mode configuration for the provisional polling feature in the `everything.cfg` file.



Push-to-talk and Group Paging

The Push-to-talk (PTT) and Group Paging features are supported on all Spectralink 84-Series handset models.

The Group Paging feature enables pages —one-way audio announcements — to users subscribed to a page group. Paging mode was originally intended primarily for desktop phones but has some use for Wi-Fi handsets that may or may not also be using PTT. In Page mode, announcements play only through the handset's speakerphone.

The Push-to-talk (PTT) feature is a collaborative tool that enables users to exchange radio broadcasts to other users subscribed to a PTT `channel`. In PTT mode, the handset behaves like a walkie-talkie; users can broadcast audio to a PTT channel and recipients subscribed to that channel can respond to your message. PTT broadcasts can be transmitted using the handset, headset, or speakerphone. They can be rejected, placed on hold and ended at any time. PTT broadcasts can be received on the speakerphone, handset, and headset.

Administrators must enable Paging and PTT before users can subscribe to a page group or PTT channel. You can enable one of these features or you can operate both simultaneously. Paging and PTT each have 25 groups/channels you can enable.

Note that you can enter a display name for sent PTT broadcasts in `ptt.displayName` and for sent page announcements in `ptt.pageMode.displayName`.



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/Later 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'. If you are deploying Spectralink 87-Series handsets and need to maintain PTT compatibility, this parameter must be set to 0 as the 87-Series handset does not support the G.726 codec.

Table 8-45: Enable Push-to-talk

Parameter	Permitted Values	Default
ptt.pttMode.enable	0 or 1	0
If 0, push-to-talk is disabled. If 1, push-to-talk is enabled.		
ptt.address	multicast IP address	224.0.1.116
The multicast IP address to send page audio to and receive page audio from. Use default.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
ptt.compatibilityMode	0 or 1	1
<p>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.</p> <p>If you are deploying Spectralink 87-Series handsets and need to maintain PTT compatibility, this parameter must be set to 0 as the 87-Series handset does not support the G.726 codec.</p>		
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.payloadSize	10 to 80	20
The audio payload size in milliseconds. Use default.		
ptt.priorityChannel	1 to 25	24
The channel assigned for priority broadcasts.		
ptt.emergencyChannel	1 to 25	25
The channel assigned for emergency broadcasts.		
ptt.channel.x.available	0 or 1	1
Make the channel available to the user		
ptt.channel.x.allowTransmit	0 or 1	1
Allow outgoing broadcasts on the channel		
ptt.channel.x.allowReceive	0 or 1	1
Allow incoming broadcasts on the channel		
ptt.channel.x.label	string	ch1: All, ch24: Priority, ch25: Emergency, others: Null
The label to identify the channel		
ptt.channel.x.subscribed	0 or 1	ch1, 24, 25: 1, others: 0
Subscribe the handset to the channel		
<p>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.</p> <p>If <code>allowTransmit</code> is disabled (0), the user cannot send PTT broadcasts on the channel. If enabled, the user may choose to send PTT broadcasts on the channel.</p> <p>If <code>allowReceive</code> is disabled (0), the user cannot receive PTT broadcasts on the channel. If enabled, the user may choose to receive PTT broadcasts on the channel.</p> <p>If <code>subscribed</code> is disabled, the handset will not be subscribed to the channel. If enabled, the handset will subscribe to the channel.</p>		

Table 8-46: 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
<p>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.</p>		

Parameter	Permitted Values	Default
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.		

Table 8-47: Additional Push-to-talk parameters

Parameter	Permitted Values	Default
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.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.		
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.		

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 8-48: Group Paging Parameters

Parameter	Permitted Values	Default
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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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	Null
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.		
ptt.pageMode.group.x.available	0 or 1	1
Make the group available to the user		
ptt.pageMode.group.x.allowTransmit	0 or 1	1
Allow outgoing announcements to the group		
ptt.pageMode.group.x.allowReceive	0 or 1	1
Allow receipt of announcements to the group		
ptt.pageMode.group.x.label	string	ch24: Priority, ch25: Emergency, others: Null
The label to identify the group		
ptt.pageMode.group.x.subscribed	0 or 1	ch1, 24, 25: 1, others: 0
Subscribe the handset to the group		
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>allowReceive</code> is disabled (0), the user cannot receive incoming pages to the group. If enabled, the user may receive incoming pages. If <code>subscribed</code> is disabled, the handset will not be subscribed to the group. If enabled, the handset will subscribe to the group.		
ptt.pageMode.payloadSize	10, 20, ..., 80 milliseconds	20
The page mode audio payload size.		
ptt.pageMode.priorityGroup	1 to 25	24
The paging group to use for priority pages.		
ptt.pageMode.transmit.timeout.continuation	0 to 65535	60
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 softkey will display on the handset. Pressing the Extend softkey continues the initial timeout for the time specified by this parameter. If 0, announcements cannot be extended.		
ptt.pageMode.transmit.timeout.initial	0 to 65535	0
The number of seconds to wait before automatically terminating an outgoing page announcement. If 0, page announcements will not automatically terminate.		

Quick Barcode Connector Application

If you are using Spectralink 8450/8452 handsets, the Spectralink Quick Barcode Connector™ (QBC) application enables you to capture and decode barcode patterns with the handset 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). Please refer to the *Quick Barcode Connector Administration Guide* for complete information.

Registrations

Each registration can optionally be associated with a private array of servers for completely segregated signaling. The Spectralink handsets support six registrations.

In the following tables, x is the registration number which can be from 1-6.

Table 8-49: Registration Parameters

Parameter	Permitted Values	Default
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@Spectralink.com) of the registration SIP URI.		
reg.x.applyServerDigitMapLocally	0 or 1	0
If 1 and <code>reg.x.server.y.specialInterop</code> is set to <code>lync2010</code> or <code>lync2013</code> , the handset 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.password	string	Null
The password to be used for authentication challenges for this registration. If the password is non-Null, it will overrule the password entered into the Authentication submenu on the Settings menu of the handset.		
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 overrule the user parameter entered into the Authentication submenu on the Settings menu of the handset.		
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> or <code>lync2013</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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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.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 2013 or 2010 Server. When deploying both 84-Series and 87-Series handsets in the same facility using the same Wireless LAN, Wi-Fi Multimedia Admission Control (aka access control, AC or WMM-AC) must be disabled in any handset parameters and APs as it is not supported by 87 Series handsets. Any parameter that requires or enforces AC must be disabled.		
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 6	1
Specify the number of line keys to use for a single registration. The maximum number of line keys is 6.		
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 handset is registered to Microsoft Lync Server 2013 or 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 handset sends all requests.		
reg.x.outboundProxy.port	0, 1 to 65535	0
The port of the SIP server to which the handset sends all requests.		
reg.x.outboundProxy.transport	DNSNaptr, TCPpreferred, UDPOnly, TLS, TCPOnly	DNSNaptr
The transport method the handset uses to communicate with the SIP server. 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. TCPpreferred – TCP is the preferred transport, UDP is used if TCP fails. UDPOnly – only UDP will be used. TLS – if TLS fails, transport fails. Leave port field empty (will default to 5061) or set to 5061. TCPOnly – only TCP will be used.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
reg.x.proxyRequire	string	Null
The string that needs to be entered in the <i>Proxy-Require</i> header. If Null, no <i>Proxy-Require</i> will be sent.		
reg.x.ringType.privateLine	default, ringer1 to ringer24	default
The ringer to be used for calls received by a private line connected to Microsoft Lync Server 2013 or 2010.		
reg.x.serverAutoDiscovery	0 or 1	1
Determines whether or not to discover the server address automatically. This parameter is used with Microsoft Lync Server 2013 or 2010.		
reg.x.serverFeatureControl.cf¹	0 or 1	0
If 0, server-based call forwarding is not enabled. If 1, server based call forwarding is enabled. This parameter overrides <code>voIpProt.SIP.serverFeatureControl.cf</code> .		
reg.x.serverFeatureControl.dnd¹	0 or 1	0
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> .		
reg.x.serverFeatureControl.localProcessing.cf	0 or 1	1
If 0 and <code>reg.x.serverFeatureControl.cf</code> is set to 1, the handset will not perform local Call Forward behavior. If set to 1, the handset 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 handset will not perform local DND call behavior. If set to 1, the handset will perform local DND call behavior on all calls received. This parameter overrides <code>voIpProt.SIP.serverFeatureControl.localProcessing.dnd</code> .		
reg.x.serverFeatureControl.signalingMethod	subscribeAsFeatureEvent, inviteFACSubscribePresence, serviceMsForwardContact	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 handset 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> .		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
reg.x.strictLineSeize	0 or 1	0
If 1, the handset is forced to wait for 200 OK on registration x when receiving a TRYING notify. This parameter overrides <code>voIpProt.SIP.strictLineSeize</code> for registration x.		
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 for IM with Microsoft Office Communications Server 2007 R2 or Microsoft Lync 2013 or 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.		

¹ Change causes handset to restart or reboot.

Multiple Registrations

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 overrule the parameters specified in `voIpProt.server.*`. The server registration parameters are listed in the following table:

Table 8-50: Registration Server Parameters

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
reg.x.server.y.address	dotted-decimal IP address or hostname	Null
The IP address or host name of a SIP server that accepts registrations. If not Null, all of the parameters in this table will overrule the parameters specified in <code>voIpProt.server.*</code> . <i>Notes:</i> If this parameter is set, it will overrule 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> .		
reg.x.server.y.expires	positive integer, minimum 10	3600
The handset's requested registration period in seconds. <i>Note:</i> The period negotiated with the server may be different. The handset 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 handset will re-register after 295 seconds (300–5).		
reg.x.server.y.expires.lineSeize	0 to 65535	30
Requested line-seize subscription period.		
reg.x.server.y.expires.overlap	5 to 65535	60
The number of seconds before the expiration time returned by server x at which the handset should try to re-register. The handset 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.		
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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
reg.x.server.y.useOutboundProxy Specify whether or not to use the outbound proxy specified in <code>reg.x.outboundProxy.address</code> for server x. This parameter overrules <code>voIpProt.server.x.useOutboundProxy</code> for registration x.	0 or 1	1
reg.x.server.y.port The port of the sip server that specifies registrations. If 0, the port used depends on <code>reg.x.server.y.transport</code> .	0, 1 to 65535	0
reg.x.server.y.register If 0, calls can be routed to an outbound proxy without registration.	0 or 1	1
reg.x.server.y.registerRetry.baseTimeOut 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.	10 - 120	60
reg.x.server.y.registerRetry.maxTimeOut 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.	60 - 1800	60
reg.x.server.y.retryMaxCount If set to 0, 3 is used. The number of retries that will be attempted before moving to the next available server.	0 to 20	3
reg.x.server.y.retryTimeOut The amount of time (in milliseconds) to wait between retries. If 0, use standard RFC 3261 signaling retry behavior.	0 to 65535	0
reg.x.server.y.specialInterop Specify if this registration should support Microsoft Office Communications Server 2007 R2 (ocs2007r2), Microsoft Live Communications Server 2005 (lcs2005), or Microsoft Lync 2010 or 2013 (lync2010 or lync2013). <i>Note:</i> To use instant messaging on Spectralink handsets, set this parameter to ocs2007r2.	standard, ocs2007r2, lcs2005, lync2010, lync2013	standard
reg.x.server.y.transport The transport method the handset uses to communicate with the SIP server. Null or DNSnaptr - 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. TCPpreferred - TCP is the preferred transport; UDP is used if TCP fails. UDPOnly - only UDP will be used. TLS - if TLS fails, transport fails. Leave port field empty (will default to 5061) or set to 5061. TCPOnly - only TCP will be used.	DNSnaptr, TCPpreferred, UDPOnly, TLS, TCPOnly	DNSnaptr

Multiple Concurrent Calls

You can enable each registered phone line to support multiple concurrent calls and have each concurrent call display on the handset'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 the next tables, you can set the maximum number of concurrent calls per registered line and the default number of calls per line key.

Summary

Parameter	Used to:
call.callsPerLineKey	Set the default number of concurrent calls for all line keys
reg.x.callsPerLineKey	Overrule the default number of calls per line key for a specific line

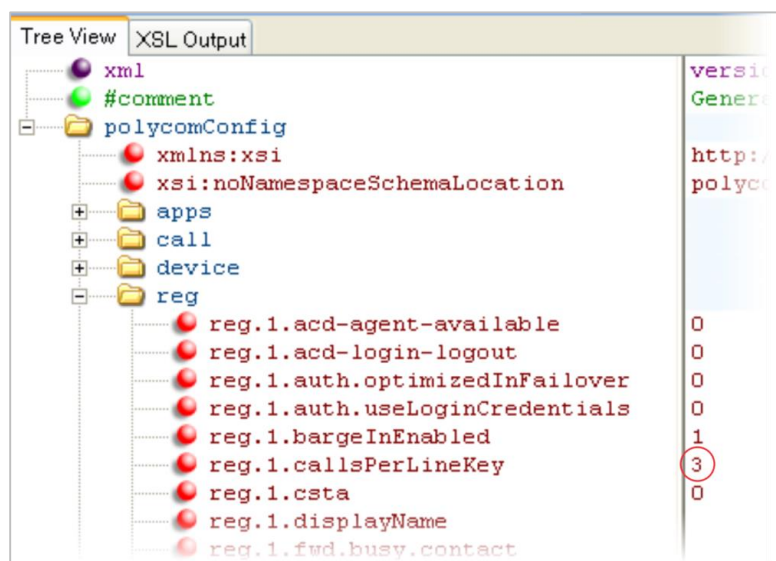
Table 8-51: Enabling Multiple Call Appearances

Parameter	Permitted Values	Default
call.callsPerLineKey	1-24	24
Set the maximum number of concurrent calls per line key. This parameter applies to all registered lines. Note that this parameter may be overruled by the per-registration parameter of <code>reg.x.callsPerLineKey</code> .		
reg.x.callsPerLineKey¹	1-24	24
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 handsets sharing that registration. This parameter overrules <code>call.callsPerLineKey</code> .		

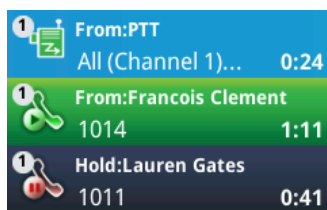
¹ Change causes handset to restart or reboot.

Example Multiple Call Appearances Configuration

The following illustration shows parameters in the `everything.cfg` file. It shows how you can enable line 1 on your handset 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 handset's screen as shown next.



Flexible Call Appearances

A number of features are associated with *Flexible Call Appearances*. Use the following information 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

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.)

<i>Phone Model</i>	Spectralink 84xx
<i>Registrations</i>	6
<i>Line Keys</i>	6
<i>Calls Per Line Key</i>	24
<i>Concurrent Calls*</i>	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.

User Profiles

User Profiles are designed to be used when handsets are shared by several users such as in shift situations. Users log on to any handset with a username and password and the handset uses their personal settings. The default password is **123**.

If a user changes any settings while logged in to a handset, the settings will be saved in an override file and loaded onto the phone the next time the same user logs in to a handset. When a user logs out, the user's personal handset settings are no longer loaded on the phone.

If the User Profile feature is set up on your company's handsets, users can:

- Log in to a handset to access their personal handset settings.
- Log out of a handset after they finish using it. (with or without a password)
- Place a call to an authorized number from a handset 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 handset. If the User Profile feature is enabled, but not required, users can choose to use the handset as is (that is, without access to their personal settings), or they can log in to have access to their personal settings. You can specify if a user is logged out of the handset when the handset restarts or reboots, or if they remain logged in.

You can also choose to define default credentials for the handset. If you specify a default user ID and password, the handset automatically logs itself in each time an actual user logs out or the handset restarts or reboots. When the handset logs itself in using the default login credentials, a default handset profile is displayed (as defined in the handset'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.



Resetting a User's Password

Spectralink recommends that you create a single default user password for all users. You can restore this default password by removing the password parameter from the override file. This will cause the handset to use the default password in the `<user>.cfg` file.

Summary

<i>Parameter</i>	<i>Used to:</i>
prov.login.enabled	Enable or disable the user profile feature
prov.login.automaticLogout	Specify the amount of time before a non-default user is logged out
prov.login.defaultPassword	Specify the default password for the default user
prov.login.defaultOnly	Specify if the handset can have users other than the default user
prov.login.defaultUser	Specify the name of the default user
prov.login.localPassword	Specify the password used to validate the user login
prov.login.persistent	Specify if a user should remain logged in after the handset reboots
prov.login.required	Specify if a user must log in while the feature is enabled
settingsLock.userProfileLogoutPassword	Specify a logout password

Table 8-52: User Profile Settings

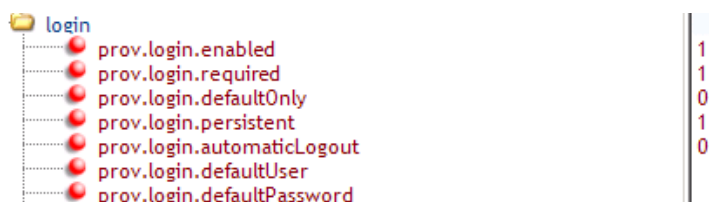
<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
prov.login.enabled	0 or 1	0
If 0, the user profile feature is disabled. If 1, the user profile feature is enabled.		
prov.login.defaultPassword	String	Null
The login password for the default user.		
prov.login.defaultOnly	0 or 1	0
If 1, the default user is the only user who can log in. If 0, other users can log in.		

Parameter	Permitted Values	Default
prov.login.defaultUser	String	Null
The username for the default user. If present, the user is automatically logged in when the handset boots up and logged in after another user logs out.		
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 handset reboots. Set this parameter to "1" if PTT parameters are specified in user-profile-specific cfg files. This way, if the phone reboots, the PTT settings persist. PTT settings (and other settings) will persist until the user logs out.		
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 cleared. If 0, the login credential password is not cleared.		
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.		
settingsLock.userProfileLogoutPassword	String [1-32 characters]	0
Logout password. When set, and the phone is configured for user profiles, the phone will: 1. require a password to access the following menu: Settings> Feature Settings> User Login> Logout This means when the user selects "Logout" they will be prompted for the password set in the above config item. 2. the Logout item on the Feature softkey will be greyed out - this will be unusable, it will not prompt for a password.		

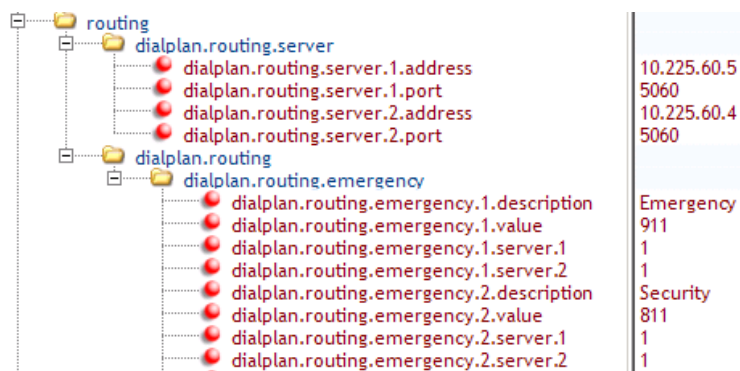
Placing Authorized (Emergency) Calls without Logging In

You may wish to allow handsets that normally require a user to provide log in credentials to be used for emergency dialing purposes without requiring the caller to log in. You can provision a list of authorized calls that can be made without logging in. The required parameters are detailed in the [Local Digit Map](#) section in the chapter on Special Use Cases and illustrated below. Also see [Emergency Dial via Authorized Call menu](#) for additional information about the Authorized Call menu.

- 1 Enable the prov.login parameter **prov.login.required**. When enabled, the handset will continuously display the **User Login** screen as shown below.



- 2 Configure the requisite parameters for emergency dial routing.



Parameters for enabling the emergency dialing feature should be implemented in a global/site configuration file so that all handsets have the emergency dialing feature enabled.

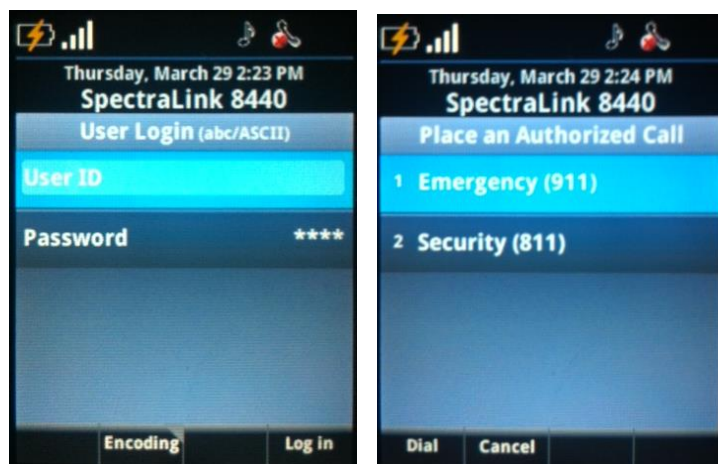


Caution: Emergency server must allow anonymous inbound SIP calls

A logged-out handset is not registered to the SIP call server. Therefore, the emergency SIP call server must be configured to allow anonymous inbound SIP calls. Otherwise the emergency calls will not be recognized.

User experience

The handset will display the login screen. When the off-hook **START** key is pressed, the user will be presented with the **Place an Authorized Call** screen enabling the user to place emergency calls.



Voicemail Integration

The Spectralink 84-Series handset is compatible with voicemail servers. You can configure each handset or line registration per handset to subscribe with a SIP URL to a voicemail server contact. You can also configure the handset to access voicemail with a single key, for example,

the **Messages** icon on the handset's Home screen. When you access the voicemail server, the handset gives a visual and audio alert; you can also configure a message waiting alert to indicate that you have unread voicemail messages.

Table 8-53: Voicemail Integration

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
up.oneTouchVoiceMail¹	0 or 1	0
If set to 1, the voicemail summary display is bypassed and voicemail is dialed directly (if configured).		
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@spectralink.com)	Null
If non-Null, the handset 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. contact – a call is placed to the contact specified by msg.mwi.x.callback. registration – the registration places a call to itself (the handset calls itself). disabled – 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@spectralink.com)	Null
The contact to call when retrieving messages for this registration if msg.mwi.x.callBackMode is set to contact. For R4.14 and higher the maximum length is 126 characters. Earlier versions are limited to 64 characters.		
up.mwiVisible¹	0 or 1	0
If set is 0, the incoming MWI notifications for lines where the MWI callback mode is disabled (msg.mwi.x.callBackMode 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.		

¹ Change causes handset to restart or reboot.

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.Spectralink.com*.

The screenshot displays the Polycom Config tool interface. The left pane shows a tree view of the configuration hierarchy. The right pane shows the corresponding XML output.

Tree View:

- xml
 - PolycomConfig
 - xsi:noNamespaceSchemaLocation
 - xmlns:xsi
 - #comment
 - #comment
 - #comment
 - #comment
 - #comment
 - Lync
 - #comment
 - openSIP
 - SIPserver
 - dialplan
 - DND_CallForwarding
 - voicemail
 - up.oneTouchVoicemail
 - up.mwiVisible
 - msg.mwi.1.callBackMode
 - msg.mwi.1.callBack
 - #comment
 - #comment
 - im

XSL Output:

```

version="1.0" encoding="utf-8" standalone="yes"

polycomConfig.xsd
http://www.w3.org/2001/XMLSchema-instance
*****
site.cfg template for FLAT DEPLOYMENT
*****
Site Parameters configured in this file apply to all...
*****

openSIP servers are basically all those that are not...

1
1
registration

In our IM example, reg2 becomes the IM line. Lync2010...
For IM, DHCP option 43 must point to the certificate...
refer to the deployment guide for manual certificate...

*****
Some common features are included below.
Feature Parameters set here will be used by all phones.
See the individual .cfg files for additional features...
*****
  
```

<volpProt/>

VoIP Protocol settings are used in wireless environments but may be overruled by <reg> parameters. You can use VoIP Protocol parameters to set up the call server and DTMF signaling.



Settings: many volpProt settings are overruled by counterpart reg.x settings

Many volpProt parameters have been replaced by reg.x parameters which will overrule their counterpart volpProt parameters if both are set. We recommend using reg.x parameters whenever possible. Some but not all volpProt parameters have replacement reg.x parameters noted in their parameter descriptions.

Table 8-54: VoIP Server Parameters

Parameter	Permitted Values	Default
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
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 overrules even if the DHCP server is available.		
volpProt.server.dhcp.type¹	0 or 1	0
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.		
volpProt.server.x.address	dotted- decimal IP address or hostname	Null
The IP address or hostname of a SIP server that accepts registrations. Multiple servers can be listed starting with x=1 to 4 for fault tolerance. Overruled by <code>reg.x.server.y.address</code> .		
volpProt.server.x.port	0, 1 to 65535	0
The port of the server that accepts registrations. If 0, the port used depends on <code>voIpProt.server.x.transport</code> . Overruled by <code>reg.x.server.y.port</code> .		
volpProt.server.x.registerRetry.baseTimeOut	10 - 120	60
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> overrules the similar <code>volpProt</code> parameter.		
volpProt.server.x.registerRetry.maxTimeOut	60 - 1800	60
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> overrules the similar <code>volpProt</code> parameter.		
volpProt.server.x.transport	DNSNaptr, TCPpreferred, UDPOnly, TLS, TCPOnly	DNSNaptr
The transport method the handset uses to communicate with the SIP server. Overruled by <code>reg.x.server.y.transport</code> values. Null or DNSNaptr – if <code>voIpProt.server.x.address</code> is a hostname and <code>voIpProt.server.x.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>voIpProt.server.x.address</code> is an IP address, or a port is given, then UDP is used. TCPpreferred – TCP is the preferred transport; UDP is used if TCP fails. UDPOnly: only UDP will be used. TLS – if TLS fails, transport fails. Leave port field empty (will default to 5061) or set to 5061. TCPOnly – only TCP will be used.		
volpProt.server.x.protocol.SIP	0 or 1	1
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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
volpProt.server.x.expires	positive integer, minimum 10	3600
The handset's requested registration period in seconds. <i>Note:</i> The period negotiated with the server may be different. The handset will attempt to re-register at the beginning of the <code>overlap</code> period. For example, if <code>expires="300"</code> and <code>overlap="5"</code> , the handset will re-register after 295 seconds (300–5). Overruled by <code>reg.x.server.y.expires</code> values.		
volpProt.server.x.expires.overlap	5 to 65535	60
The number of seconds before the expiration time returned by server x at which the handset should try to re-register. The handset 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.		
volpProt.server.x.expires.lineSeize	0 to 65535	30
Requested line-seize subscription period.		
volpProt.server.x.lcs	0 or 1	0
If 0, the Microsoft Live Communications Server (LCS) is not supported. If 1, LCS is supported for registration x. This parameter overrules <code>voIpProt.SIP.lcs</code> .		
volpProt.server.x.register	0 or 1	1
If 0, calls can be routed to an outbound proxy without registration. See <code>reg.x.server.y.register</code> .		
volpProt.server.x.retryTimeOut	0 to 65535	0
The amount of time (in milliseconds) to wait between retries. If 0, use standard RFC 3261 signaling retry behavior.		
volpProt.server.x.retryMaxCount	0 to 20	3
If set to 0, 3 is used. The number of retries that will be attempted before moving to the next available server.		
volpProt.server.x.specialInterop	standard, ocs2007r2, lcs2005, lync2010, lync2013	standard
Specify if this registration should support Microsoft Office Communications Server 2007 R2 (ocs2007r2), Microsoft Live Communications Server 2005 (lcs2005), or Microsoft Lync 2010 or 2013 (lync2010 or lync2013).		
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.		

¹ Change causes handset to restart or reboot.

Table 8-55: 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 handset uses its own preference list when deciding which codec to use rather than the preference list in an offer. If set to 0, it is disabled.		
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.		
volpProt.SDP.useLegacyPayloadTypeNegotiation	0 or 1	0
If set to 1, the handset transmits and receives RTP using the payload type identified by the first codec listed in the SDP of the codec negotiation answer.		

Parameter	Permitted Values	Default
If set to 0, RFC 3264 is followed for transmit and receive RTP payload type values.		

Table 8-56: Session Initiation Protocol (SIP) Parameters

Parameter	Permitted Values	Default
volpProt.SIP.acd.signalingMethod¹	0 or 1	0
If set to 0, the 'SIP-B' signaling is supported. (This is the older ACD functionality.) If set to 1, the feature synchronization signaling is supported. (This is the new ACD functionality.)		
volpProt.SIP.CID.sourcePreference	ASCII string up to 120 characters long	Null
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 From, P-Asserted-Identity, Remote-Party-ID and P-Asserted-Identity, Remote-Party-ID are also valid.		
volpProt.SIP.compliance.RFC3261.validate.contentLanguage	0 or 1	1
If set to 1, validation of the SIP header content language is enabled. If set to 0, validation is disabled.		
volpProt.SIP.compliance.RFC3261.validate.contentLength	0 or 1	1
If set to 1, validation of the SIP header content length is enabled. If set to 0, validation is disabled.		
volpProt.SIP.compliance.RFC3261.validate.uriScheme	0 or 1	1
If set to 1, validation of the SIP header URI scheme is enabled. If set to 0, validation is disabled.		
volpProt.SIP.conference.address	ASCII string up to 128 characters long	Null
If Null, conferences are set up on the handset locally. 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.		
volpProt.SIP.conference.parallelRefer	0 or 1	0
If 1, a parallel REFER is sent to the call server. Note: This parameter must be set for Siemens Openscape Centralized Conferencing.		
volpProt.SIP.connectionReuse.useAlias	0 or 1	0
If set to 0, this is the old behavior. If set to 1, handset uses the connection reuse draft which introduces "alias".		
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 overrule this parameter).		
volpProt.SIP.dialog.strictXLineID	0 or 1	0
If 0, the handset 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 handset 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, handset uses a <code>pval</code> field name in the Dialog. This obeys the draft-ietf-sipping-dialog-package-06.txt draft. If set to 1, the handset uses a field name of <code>pvalue</code> .		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
volpProt.SIP.dialog.useSDP 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.	0 or 1	0
volpProt.SIP.dtmfViaSignaling.rfc2976¹ 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.	0 or 1	0
volpProt.SIP.enable¹ 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.	0 or 1	1
volpProt.SIP.IM.autoAnswerDelay The time interval in seconds from receipt of the instant message invitation to automatically accepting the invitation.	0 to 40	10
volpProt.SIP.keepalive.sessionTimers If set to 1, the session timer will be enabled. If set to 0, the session timer will be disabled, and the handset will not declare "timer" in "Support" header in an INVITE. The handset will still respond to a re-INVITE or UPDATE. The handset will not try to re-INVITE or UPDATE even if the remote end point asks for it.	0 or 1	0
volpProt.SIP.lineSeize.retries Controls the number of times the handset will retry a notify when attempting to seize a line (BLA).	3 to 10	10
volpProt.SIP.local.port¹ 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.	0 to 65535	5060
volpProt.SIP.mtls.enable If 0, TLS with mutual authentication is disabled. If 1, TLS with mutual authentication is enabled. Used in conjunction with Microsoft Lync 2013 or 2010.	0 or 1	1
volpProt.SIP.outboundProxy.address The IP address or hostname of the SIP server to which the handset sends all requests.	dotted-decimal IP address or hostname	Null
volpProt.SIP.outboundProxy.port The port of the SIP server to which the handset sends all requests.	0 to 65535	0
volpProt.SIP.outboundProxy.transport The transport method the handset uses to communicate with the SIP server. Null or DNSNaptr – if reg.x.outboundProxy.address is a hostname and reg.x.outboundProxy.port is 0 or Null, do NAPTR then SRV look-ups to try to discover the transport, ports and servers, as per RFC 3263. If reg.x.outboundProxy.address is an IP address, or a port is given, then UDP is used. TCPpreferred – TCP is the preferred transport, UDP is used if TCP fails. UDPOnly – only UDP will be used. TLS – if TLS fails, transport fails. Leave port field empty (will default to 5061) or set to 5061. TCPOnly – only TCP will be used.	DNSNaptr, TCPpreferred, UDPOnly, TLS, TCPOnly	DNSNaptr
volpProt.SIP.pingInterval The number in seconds to send "PING" message. This feature is disabled by default.	0 to 3600	0

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
volpProt.SIP.pingMethod The ping method to be used.	PING, OPTIONS	PING
volpProt.SIP.presence.nortelShortMode¹ 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 handset.	0 or 1	0
volpProt.SIP.requestValidation.digest.realm¹ Determines the string used for Realm.	wirelessIpPhone	wirelessIpPhone
volpProt.SIP.requestURI.E164.addGlobalPrefix If set to 1, '+' global prefix is added to the E.164 user parts in sip: URIs.	0 or 1	0
volpProt.SIP.sendCompactHdrs If set to 0, SIP header names generated by the handset use the long form, for example <code>From</code> . If set to 1, SIP header names generated by the handset use the short form, for example <code>f</code> .	0 or 1	0
volpProt.SIP.serverFeatureControl.dnd 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.	0 or 1	0
volpProt.SIP.serverFeatureControl.missedCalls¹ 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.	0 or 1	0
volpProt.SIP.serverFeatureControl.localProcessing.dnd If set to 0 and <code>volIpProt.SIP.serverFeatureControl.dnd</code> is set to 1, the handset will not perform local DND call behavior. If set to 1, the handset will perform local DND call behavior on all calls received.	0 or 1	1
volpProt.SIP.specialEvent.checkSync.alwaysReboot¹ If set to 1, always reboot when a NOTIFY message is received from the server with event equal to check-sync. If set to 0, only reboot if any of the files listed in <code><MAC-address>.cfg</code> have changed on the FTP server when a NOTIFY message is received from the server with event equal to check-sync.	0 or 1	0
volpProt.SIP.specialEvent.lineSeize.nonStandard¹ 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.	0 or 1	1
volpProt.SIP.strictLineSeize If set to 1, The handset is forced to wait for a 200 OK response when receiving a TRYING notify. If set to 0, this is old behavior.	0 or 1	0
volpProt.SIP.strictUserValidation If set to 1, the handset is forced to match the user portion of signaling exactly. If set to 0, the handset will use the first registration if the user part does not match any registration.	0 or 1	0
volpProt.SIP.tlsDsk.enable If 0, TLS DSK is disabled. If 1, TLS DSK is enabled.	0 or 1	0
volpProt.SIP.turnOffNonSecureTransport¹ If set to 1, stop listening to port 5060 when using AS-SIP enabled.	0 or 1	0

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
volpProt.SIP.use486forReject If set to 1 and the handset is indicating a ringing inbound call appearance, the handset will transmit a 486 response to the received INVITE when the Reject softkey is pressed. If set to 0, no 486 response is transmitted.	0 or 1	0
voipPort.SIP.useCompleteUriForRetrieve 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.	0 or 1	1
volpProt.SIP.useContactInReferTo If set to 0, the "To URI" is used in the REFER. If set to 1, the "Contact URI" is used in the REFER.	0 or 1	0

¹ Change causes handset to restart or reboot.

Web Browser

There are two aspects to the web browser. One is the list of apps that are accessible from the Applications icon. The other is the development of these apps and the parameters that are used to implement them.

The Spectralink handsets support a full Web browser that the user can launch by navigating to **Home> Applications**. The label and url for the web application is set in the parameters listed below. Other <apps/> parameters are used to control telephone notification events, state polling events, and push server controls.

For more information on how to use the control parameters, see the *Spectralink Web Application Developer's Guide*.

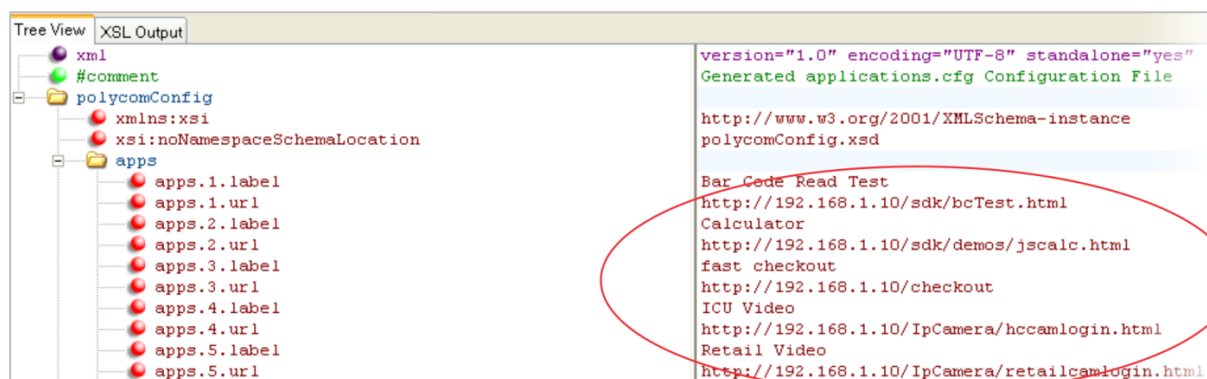
Table 8-57: Apps listed on the handset

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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).		

² For the Spectralink 84-Series handsets, the toolbar autohide is disabled by default.

Example Web Browser Configuration

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.

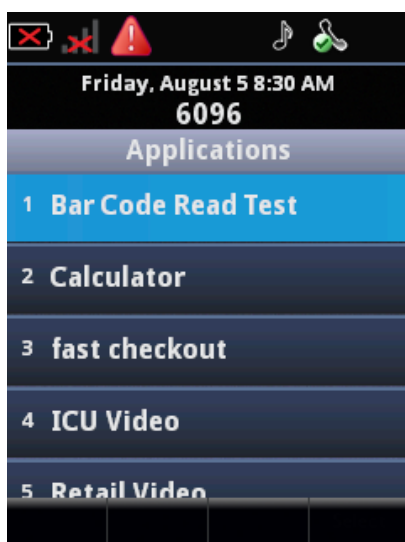


Table 8-58: SDK Application Parameters

Parameter	Permitted Values	Default
apps.telNotification.heartbeatTimeoutSeconds	0 – 65,535	0
Implements a "heartbeat" timer for telephony notifications that repeats the line registration notification on a periodic basis if set. The default value for this parameter is 0, where the timer does not start at all, so there is no change to the behavior unless the apps.telNotification.heartbeatTimeoutSeconds parameter is set to a non-zero value. With a heartbeat set, when the phone goes out of range or is power cycled, notifications are not disrupted.		
apps.push.messageType	0 to 5	0
Choose a priority level for push messages from the application server to the handset. 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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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. To disable the non-secure push (HTTP) requires that both apps.push.secureTunnelEnabled and apps.push.secureTunnelRequired be set to 0.		
apps.push.secureTunnelPort	1 to 65535	443
The port that the handset 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. To disable the non-secure push (HTTP) requires that both apps.push.secureTunnelEnabled and apps.push.secureTunnelRequired be set to 0.		
apps.push.serverRootURL	URL	null
The URL of the application server you enter here is combined with the handset address and sent to the handset's browser. For example, if the application server root URL is http://172.24.128.85:8080/sampleapps and the relative URL is /examples/sample.html, the URL that is sent to the browser is http://172.24.128.85:8080/sampleapps/examples/sample.html. 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 apps.push.username and apps.push.password must be set (not null).		
apps.statePolling.password	string	null
Enter the password that the handset requires to authenticate handset state polling.		
apps.statePolling.URL	URL	null
The URL to which the handset 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 apps.statePolling.URL, apps.statePolling.username, and apps.statePolling.password must be set to non-null values.		
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 handset requires to authenticate handset 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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
apps.telNotification.outgoingEvent	0 or 1	0
If 0, outgoing call notification is disabled. If 1, notification is enabled.		
apps.telNotification.scanBarcodeEvent	0 or 1	0
If 0, event 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 handset sends notifications of specified events. Can be either HTTP or HTTPS.		
apps.telNotification.x.URL	URL	null
The URL to which the handset 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. The UserLoginOut Event can be used to detect when a user signs into or out of a phone using a profile. A notification will also be sent whenever the phone is power cycled, whether or not the same login is used (i.e. persistent). This way you can verify who is using the phone through a power cycle as well as any login/logout cycle.		

<mb/>

This parameter's settings control the home page, proxy and size limits to be used by the browser when it is selected to provide services.

Table 8-59: Web Browser Parameters

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
mb.main.autoBackKey¹	0 or 1	1
If 0, the handset does not provide a Back softkey; all softkeys are created and controlled by the application. If 1, the handset automatically supplies a Back softkey in all main browser screens. The Back softkey will take the user back to the previous page in the browser history.		
mb.main.home	Any fully formed valid HTTP URL. Length up to 255 characters.	Null
The URL of the browser'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 to 600	40
The timeout, in seconds, for the interactive browser. If the interactive browser remains idle for the defined period of time, the handset returns to idle. If 0, there is no timeout.		
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 browser. If blank, normal unproxied HTTP is used by the browser.		
mb.main.toolbar.autoHide.enabled	0 or 1	1
If 0, the toolbar displays continually. If 1, the toolbar disappears if not selected.		

¹ Change causes handset to restart or reboot.

<oai/>

The Spectralink handsets support communications using the Open Application Interface (OAI). Also see [Open Application Interface](#). You can set the connection parameters using the table shown next:

Table 8-60: Open Application Interface (OAI) Parameters

<i>Parameter</i>	<i>Permitted values</i>	<i>Default</i>
oai.gateway.address The address of the OAI server.	IP address	Null
oai.userId 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.	String of eight hexadecimal characters	Null

Chapter 9: Web Application Parameters

Web application parameters allow the handset to work with an application, providing information, alerts and notifications through configuration of parameters that are designed for interfacing with third party applications. Consult with your application developer for parameter requirements.

The parameters described in this chapter include those for:

- Web applications
- Push requests
- Telephone event notifications
- Phone state polling

A sample configuration is shown in [Sample Configurations](#).

Application menu configuration <apps.>

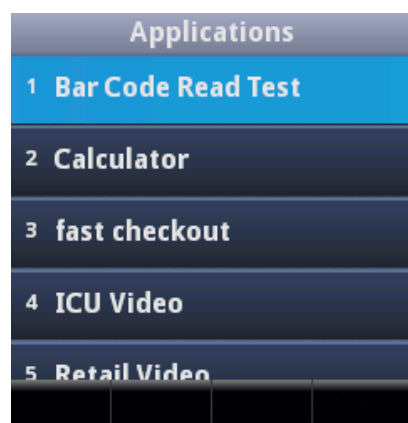
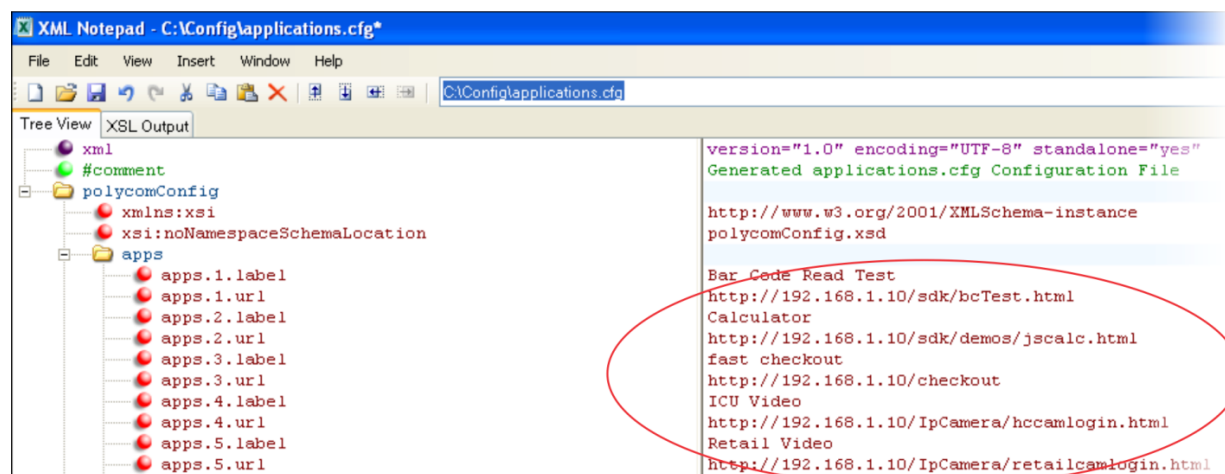
Table 9-1: Apps listed on the phone

Parameter	Permitted Values	Default
apps.x.label ² The descriptive text that displays in the Applications menu	String	null
apps.x.url ² The URL of an application		
The label and URL of up to 12 applications (for x = 1 to 12).		

² For the Spectralink 84-Series handsets, the toolbar autohide is disabled by default.

Example Web browser Configuration

The following example illustrates the use of these two parameters and how they look on the Application menu.



Web browser parameters <mb.>

The <mb.> parameters control the home page, proxy and size limits to be used by the browser when it is selected to provide services.

Table 9-2: Web browser Parameters

Parameter	Permitted Values	Default
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.		
mb.main.home	Any fully formed valid HTTP URL. Length up to 255 characters.	Null
The URL of the browser's Home page. For example: <i>http://www.example.com/xhtml/frontpage/home</i> . If no apps.x.url applications are configured, and mb.main.home is configured, then when the user selects the Applications icon from the carousel, the mb.main.hone will be immediately loaded.		

Parameter	Permitted Values	Default
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.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 browser. If blank, normal unproxied HTTP is used by the browser.		

¹ Change causes phone to restart or reboot.

State Polling Parameters <apps.statePolling.>

The <apps.statePolling/> parameter is used to control state polling events.

Table 9-3: Telephone Event Notification Parameters

Parameter	Permitted Values	Default
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. Note: 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.		
apps.statePolling.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.		

Push Request Parameters <apps.push.>

The <apps.push/> parameters are used to control push server controls.



Settings: Enabling Data URL Push

Both `apps.push.username` and `apps.push.password` must be set for Data URL Push to be enabled.



Note: Parameter available in Release 4.5, 4.7 and later

The expanded `np.normal.alert.x` parameters have been implemented in Release 4.5.x, 4.7.x and later.

Table 9-4: Push Parameters

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
np.normal.alert.apiPush.tonePattern	silent, messageWaiting, instantMessage, remoteHoldNotification, localHoldNotification, positiveConfirm, negativeConfirm, welcome, misc1, misc2, misc3, misc4, misc5, misc6, misc7, custom1, custom2	silent
If silent, there is no sound when an alert is pushed.		
np.normal.alert.apiPush.vibration	0 or 1	0
If 0, there is no sound when an alert is pushed. If 1, the handset's selected ring profile's configured API Push alert tone is played.		
np.silent.alert.apiPush.tonePattern	0 or 1	0
If 0, there is no sound when an alert is pushed. If 1, the handset's selected ring profile's configured API Push alert tone is played.		
np.silent.alert.apiPush.vibration	0 or 1	0
If 0, there is no sound when an alert is pushed. If 1, the handset's selected ring profile's configured API Push alert tone is played.		
np.meeting.alert.apiPush.tonePattern	0 or 1	0
If 0, there is no sound when an alert is pushed. If 1, the handset's selected ring profile's configured API Push alert tone is played.		
np.meeting.alert.apiPush.vibration	0 or 1	0
If 0, there is no sound when an alert is pushed. If 1, the handset's selected ring profile's configured API Push alert tone is played.		
np.custom1.alert.apiPush.tonePattern	0 or 1	0
If 0, there is no sound when an alert is pushed. If 1, the handset's selected ring profile's configured API Push alert tone is played.		
np.custom1.alert.apiPush.vibration	0 or 1	0
If 0, there is no sound when an alert is pushed. If 1, the handset's selected ring profile's configured API Push alert tone is played.		
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 messages 5: (All) Allows all push messages See Caution: Critical message conflict note after this table.		
apps.push.password	string	null
The password to access the push server URL. Used with the username to respond to the HTTP Digest Challenge from the handset.		

Parameter	Permitted Values	Default
apps.push.play.incall.volume.scaling	-36dB to 0dB	0dB
<p>This parameter allows the administrator to attenuate the volume level of alert tones played in-ear when a user is in the call. Some alert tones may be perceived by users as being loud and this parameter allows an administrator to apply volume reduction to all tones played by the API when in call.</p> <p>If 0, the volume of the push alert is the same as the call volume. Settings of -10, -20 and -36 get progressively quieter.</p>		
apps.push.secureTunnelEnabled	0 or 1	1
<p>If 0, the Web server is not connected through a secure tunnel. If 1, the Web server is connected through a secure tunnel.</p>		
apps.push.secureTunnelPort	1 to 65535	443
<p>The port that the phone should use to communicate to the Web server when the secure tunnel is used.</p>		
apps.push.secureTunnelRequired	0 or 1	0
<p>If 0, communications to the Web server do not require a secure tunnel. If 1, communications require a secure tunnel.</p>		
apps.push.serverRootURL	URL	null
<p>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.</p>		
apps.push.username	string	null
<p>The user name to access the push server URL. Used with the password to respond to the HTTP Digest Challenge from the handset.</p> <p>Note: To enable the push functionality, the parameters <code>apps.push.username</code> and <code>apps.push.password</code> must be set (not null).</p>		



Caution: Critical message conflict

Any new critical message overlays the previous one, but when 2 critical messages come to the phone too close together and both require the phone to retrieve files, the retrieval of the first file can be cut short by the retrieval of the second. This will cause the first file to not be processed properly or not at all.

For instance, if the first critical message causes the phone to retrieve an html file with `javaScript` in it to write text on the phone's display and the second critical message causes the phone to retrieve a wav file to play a tune, the user may not see the display on the phone but will just hear the tune.

Telephony Notification Parameters (`apps.telNotification.>`)

The `<apps.telNotification/>` parameter is used to control telephone notification events.

Table 9-5: Telephone Heartbeat Parameter

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
apps.telNotification.heartbeatTimeoutSeconds	0 – 65,535	0
<p>Implements a "heartbeat" timer for telephony notifications that repeats the line registration notification on a periodic basis if set. The default value for this parameter is 0, where the timer does not start at all, so there is no change to the behavior unless the apps.telNotification.heartbeatTimeoutSeconds parameter is set to a non-zero value. With a heartbeat set, when the phone goes out of range or is power cycled, notifications are not disrupted.</p>		

Table 9-6: Telephone Event Notification Parameters

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
apps.telNotification.alarmEvent	0 or 1	0
<p>Used with Personal Alarms. If 0, alarm event notification is disabled. If 1, notification is enabled. When this parameter is enabled, an XML notification is sent when an alarm event occurs. Alarm events occur when Running, Tilt, and Still alarms go off and when Duress/Emergency Calls are made.</p>		
apps.telNotification.callStateChangeEvent	0 or 1	0
<p>If 0, call state change notification is disabled. If 1, notification is enabled.</p>		
apps.telNotification.incomingEvent	0 or 1	0
<p>If 0, incoming call notification is disabled. If 1, notification is enabled.</p>		
apps.telNotification.InChargerEvent	0 or 1	0
<p>If 0, notification is disabled. If 1, notification is enabled.</p>		
apps.telNotification.OutChargerEvent	0 or 1	0
<p>If 0, is disabled. If 1, notification is enabled.</p>		
apps.telNotification.lineRegistrationEvent	0 or 1	0
<p>If 0, line registration notification is disabled. If 1, notification is enabled.</p>		
apps.telNotification.networkUpEvent	0 or 1	0
<p>If 0, network up notification is disabled. If 1, notification is enabled.</p>		
apps.telNotification.offhookEvent	0 or 1	0
<p>If 0, off-hook notification is disabled. If 1, notification is enabled.</p>		
apps.telNotification.onhookEvent	0 or 1	0
<p>If 0, on-hook notification is disabled. If 1, notification is enabled.</p>		
apps.telNotification.outgoingEvent	0 or 1	0
<p>If 0, outgoing call notification is disabled. If 1, notification is enabled.</p>		
apps.telNotification.unInitializationEvent	0 or 1	0
<p>If 0, line unregistration notification is disabled. If 1, notification is enabled.</p>		
apps.telNotification.URL	URL	null
<p>The URL to which the phone sends notifications of specified events. Can be either HTTP or HTTPS.</p>		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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.		



Admin Tip: Limitation for server URLs

The configured events will be sent to all apps.telNotification. URL and all appstelNotification.x.URLs that are configured. There is no way you can configure a few events for a specific server and remaining events for another server.

Open Application Interface parameters <oai.>

The Spectralink handsets support communications using the Open Application Interface (OAI). You can set the connection parameters using the table shown next:

Table 9-6: 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.		
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 handset identifies itself to the OAI gateway as 00:00:ww:xx:yy:zz. The upper two bytes are zero and the lower four bytes are the oai.userId.		
oai.keyRepeatAcceleration.enabled	0, 1	1
The default is "1", enabled, which is the usual behavior where the key repeat accelerates to one keypress every 50ms. To prevent excessive messaging when the key is held down, set the parameter to "0" which reduces the key repeat rate to 500ms. This parameter only affects OAI calls.		

Sample Configurations

The following sample configurations show how configuration parameters are used in applications.

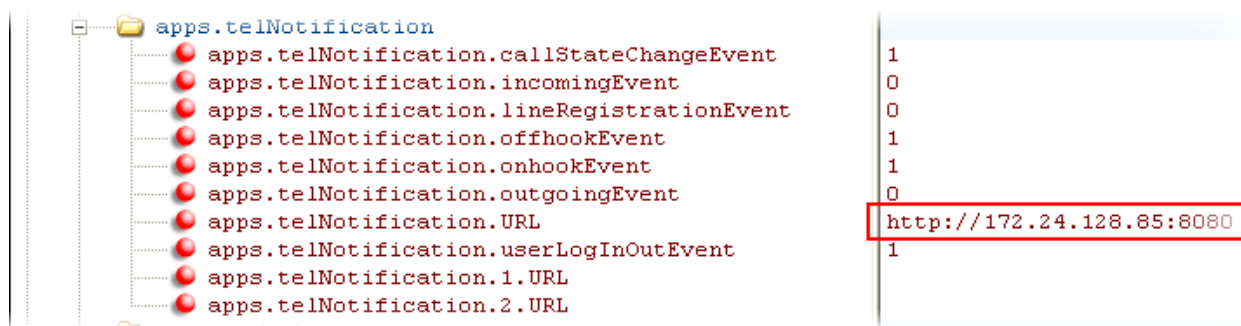
Push

- `apps.push.messageType` is set to the appropriate display priority. For example, **3** – Important Priority messages only.

- `apps.serverRootURL` is set to the application server root URL. For example, `http://172.24.128.85:8080/sampleapps`.
- `apps.push.username` is set to the appropriate user name. For example, **bob**.
- `apps.push.password` is set to the appropriate password. For example, **1234**.

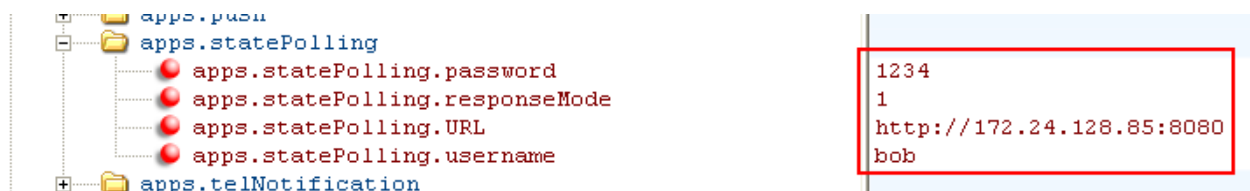
Telephony Notifications

- `apps.telNotification.URL` is set to the URL where notifications should be sent. For example, `http://172.24.128.85:8080`.
- `apps.telNotification.offhookEvent` is set to 1 to enable notifications for off-hook events.
- `apps.telNotification.onhookEvent` is set to 1 to enable notifications for on-hook events.
- `apps.telNotification.userLogInOut` is set to 1 to enable notifications for user login and logout events.
- `apps.telNotification.callStateChange` is set to 1 to enable notifications for call state change events.



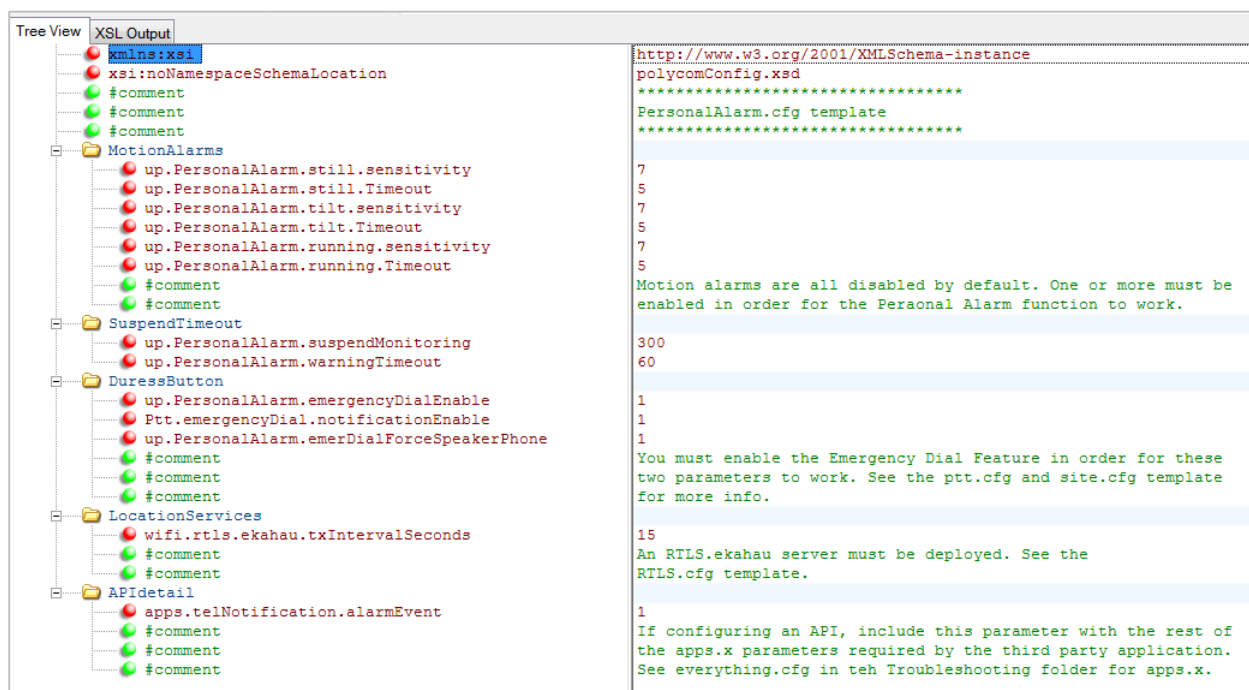
State Polling

- `apps.statePolling.URL` is set to the location where requested state polling information should be sent. For example, `http://172.24.128.85:8080`.
- `apps.statePolling.responseMode` is set to send the requested state polling information to the configured URL, 1, instead of back to the requestor.
- `apps.statePolling.username` is set to the appropriate username. For example, **bob**.
- `apps.statePolling.password` is set to the appropriate password. For example, **1234**.



Personal Alarms

The PersonalAlarm.cfg template is provided with the Spectralink software download in the Config > Features folder. Use it as a starting point to customize your deployment of this feature.



Chapter 10: System-Level Parameters

Certain parameters address system-level settings, such as DHCP.

Configuration File Encryption

You can encrypt configuration files (excluding the master configuration file), contact directories, and configuration override files. See [Encrypting Configuration Files](#) for how to encrypt configuration files. This section allow you to regulate the behavior of encrypted files.

Summary

<i>Parameter</i>	<i>Used to:</i>
sec.encryption.upload.callLists	Specify if the call list overrides file should be encrypted when it is uploaded from the handset to the server
sec.encryption.upload.config	Specify if configuration files uploaded from the handset to the provisioning server should be encrypted
sec.encryption.upload.dir	Specify if the contact directory is encrypted when it is uploaded from the handset to the provisioning server
sec.encryption.upload.overrides	Specify if the configuration overrides file should be encrypted when it is uploaded from the handset to the server
device.sec.configEncryption.key	Specify an encryption key so the handset can download encrypted files from the provisioning server.

Table 10-1: Configuration File Encryption

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
sec.encryption.upload.callLists¹	0 or 1	0
The encryption on the handset-specific call lists that is uploaded to the provisioning server. If 0, the file is uploaded unencrypted, and overwrites whatever handset-specific configuration file 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 handset-specific call lists file on the server.		
sec.encryption.upload.config	0 or 1	0
The encryption on the handset-specific configuration file created and uploaded to the provisioning server when the user selects Upload Configuration from the handset menu. If 0, the file is uploaded unencrypted, and overwrites whatever handset-specific configuration file is on the server, even if the file on the server is encrypted. If 1, the file is uploaded encrypted and replaces any existing handset-specific configuration file on the server.		
sec.encryption.upload.dir¹	0 or 1	0
The encryption on the handset-specific contact directory that is uploaded to the provisioning server. If 0, the directory is uploaded unencrypted regardless of how it was downloaded, the directory replaces whatever handset-specific contact directory is on the server, even if the file on the server is encrypted. If 1, the directory is uploaded encrypted regardless of how it was downloaded. The file replaces any existing handset-specific contact directory file on the server.		

Parameter	Permitted Values	Default
sec.encryption.upload.overrides	0 or 1	0
<p>The encryption on the handset-specific <MACaddress>-phone.cfg override file that is uploaded to the server. 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 handset-specific override file on the server.</p>		

¹ Change causes handset to restart or reboot.

Understanding Digital Certificates

Spectralink handsets are installed with a Spectralink-authenticated RSA device certificate. You can use this certificate to create a secure connection between handset and server when initiating Transport Layer Security (TLS) communications over protocols such as FTPS, HTTPS and SIP. A device certificate is used with WPA2 Enterprise EAP-TLS security method.

You can download the Spectralink Root CA from <http://pki.Spectralink.com/aia/Spectralink%20Issuing%20CA.crt>. Entering this URL exactly as shown will download the certificate to your computer (you will be prompted on whether to save or open the certificate). Note that the certificate is set to expire on March 9, 2044.



Note: How do I get the Spectralink certificates?

Spectralink CA certificates can be obtained from:

<http://pki.spectralink.com/aia/Spectralink%20Issuing%20CA.crt>

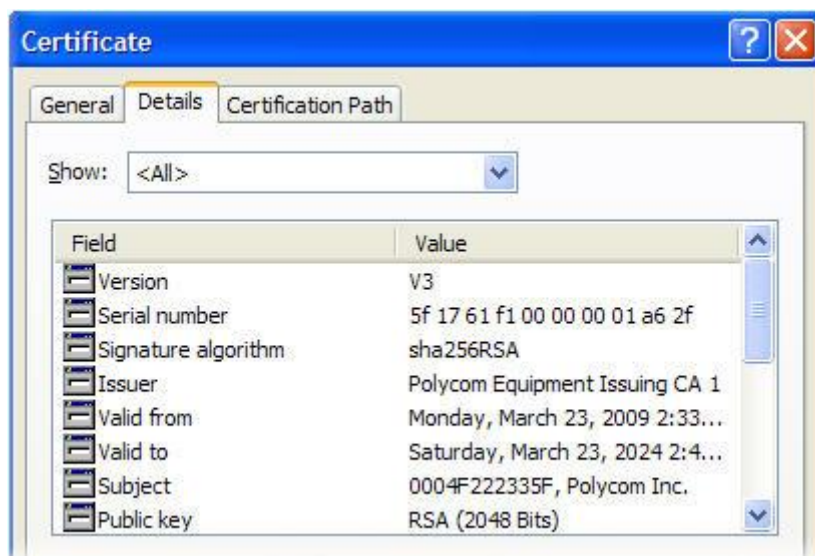
<http://pki.spectralink.com/aia/Spectralink%20Root%20CA.crt>

Spectralink 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 Spectralink device certificate when opened in Microsoft Windows.



The device certificate and associated private key are stored on the handset 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 Spectralink handsets

As of 84-Series software 4.0.0, you can install custom device certificates on your Spectralink handsets. These certificates are installed in the same way custom CA certificates are installed. See Technical Bulletin CS-13-06: *Using custom certificates with Spectralink 8400 handsets*.

To determine if there is a custom device certificate on a Spectralink handset:

Navigate to **Settings> Advanced Settings> [enter password]> Administration Settings> TLS Security> Custom Device Credentials**.

To view the status of the Spectralink device certificate on the handset:

Navigate to **Settings > Status> Platform> Phone> 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.
- Signed by:** displays the common name of the signing Certificate Authority (CA), e.g. "Spectralink".

- **Self-signed** is displayed if other certificates have been installed and the Signed by field could be a MAC address.
- **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 Spectralink handsets 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 Spectralink Root CA that issued these certificates, the handsets will operate correctly.

About Digital Certificates

Certificates enable handsets and servers to authenticate each other before permitting any exchange of data. Certificates are utilized by the Transport Layer Security (**TLS**) protocol which ensures that no third party may eavesdrop or tamper with any message. Certificates also allow for securely encrypted data to be passed between a SIP client and the PBX, (including Lync), for secure provisioning under the FTPS, HTTPS protocols, for secure browser communications and for secure syslogging.

A number of commonly-used certificates are loaded in the phone at the factory. These are called "built-in" certificates. See [Appendix E: Trusted Certificate Authority List](#) for the list of authorities.

Types of certificates

CA certificate used for 802.1x Authentication

A CA certificate can be used for 802.1x Authentication. EAP-TLS and EAP-FAST.

The client (the handset) uses the CA certificate to verify the CA signature of the Authentication server (the RADIUS server) certificate before establishing a secure connection. This way the handset knows it's talking to the correct far side. It is provided by a Certificate Authority (hence CA) or an IT administrator and must be loaded on the handset during configuration. This type of certificate is also called a server certificate as it certifies the server. The certificate is usually the same for every device and is therefore also known as a "public" certificate.

CA certificate used for Secure Syslog

A CA certificate can be used for Secure Syslog. Syslog can be configured to use a secure TLS Tunnel using the CA certificate.

CA certificate used for Secure Provisioning

A CA Certificate can be used for Secure Provisioning: FTPS and HTTPS. This prevents the configuration parameters of a device from being exposed during wireless transfer.

CA certificate used for SIP Communication

Some PBXes allow for communication between the handset and the device to occur using TLS. A CA certificate can be used to set up MTLS (mutual TLS) between the handset and the PBX in this case.

Other

CA Certificate used for Browser Communication

CA certificate used for LDAP Communication

Device certificate used for 802.1x Authentication

A device certificate validates the handset to the RADIUS server during EAP-TLS Authentication. Spectralink 84-Series handsets are shipped with a Spectralink device certificate and its associated private key known only to the phone which can be used by EAP-TLS for Wi-Fi security. The Spectralink device certificate uses the handset's MAC address as its common name which is also its Identity.

PAC file

The Protected Access Credential (PAC) is a proprietary Cisco method for provisioning certificates. The PAC can be either a specific to a device or common to a group of devices. It is generated by the RADIUS server and must be loaded either manually or automatically. EAP-FAST is used with Cisco® products and by a number of other WLAN vendors.

Configuring certificates

To configure a certificate, you will follow these general steps:

- 1** Loading the certificate
- 2** Assigning the certificate to a Platform or Application profile
- 3** Connecting the platform to how it will be used
- 4** Configuring additional parameters (if necessary)

For purposes of configuration, certificates are divided into two different categories—Platform and Application. If set, the two platform certificates are stored in the device's flash memory and are used by both the Updater and the application parts of the software. If any are set, the six application certificates are stored in the device's RAM and are used by the application part of the software.

Platform certificates

Platform types are designed to be used for parameters that are required for initial wireless configuration and provisioning. Their configuration parameters begin with `device`. Two Platform certificate “slots” are available. Platform profiles are used by:

- 802.1x (parameters are called “dotx” and refer to Wi-Fi security methods)
- Provisioning server transport protocol
- Syslog server

#1 Loading Platform certificates

You will load the Platform certificates using the below parameters in the `wireless.cfg` file. Spectralink recommends that you load Wi-Fi certificates in slot1 and provisioning/syslog server certificates in slot2. This parallels where SLIC loads certificates for the Wi-Fi security method and for the provisioning server.

- `device.sec.TLS.customCaCert1` “slot1”
- `device.sec.TLS.customCaCert2` “slot2”

#2 Assigning the certificates to a Profile List

You can broadly or narrowly define which certificates in the handset's certificate store can be used for authentication purposes. There are two profile lists, `CaCertList1` and `CaCertList2`. The default is `All` which means the handset will look through all certs loaded on the phone to locate the one it needs for a particular use. The options are `Platform1` or `Platform2` or various combinations.

`All` references any loaded certificate. `All` is the default.

`Platform1` references the certificate loaded into `device.sec.TLS.customCaCert1`

`Platform2` references the certificate loaded into `device.sec.TLS.customCaCert2`

`Platform1AndPlatform2` references either one of the two platform certs.

`Builtin` references the large number of well-known certificate authorities pre-loaded into the handset.

Instead of having the application search through all the certificates, you can use the `CaCertList` parameter to point the application to the exact certificate(s) you want it to use which speeds up the process and makes bringing up the handsets more efficient. Spectralink recommends assigning `Platform1` to `CaCert1` and `Platform2` to `CaCert2`. Use the following parameters to configure the Profile Lists:

- Profile List 1 is configured with `device.sec.TLS.profile.CaCertList1`
- Profile List 2 is configured with `device.sec.TLS.profile.CaCertList2`

#3 Connect the Platform to a use

In this step you assign the program that is using the certificate(s) to a Profile List that you created above. More than one program can use a certificate. Spectralink recommends using Platform 1 or Platform 2 for Platform uses; provisioning server, syslog server and Wi-Fi security:

- Provisioning server: `device.sec.TLS.profileSelection.provisioning`
- Wi-Fi security: `device.sec.TLS.profileSelection.dot1x`
- Syslog server: `device.sec.TLS.profileSelection.syslog`

Example: 802.11X with EAP-TLS

How the parameters are used to set Wi-Fi security method EAP-TLS.

```
<WPA2Enterprise.EAP-TLS
  device.wifi.securityMode="WPA2-Enterprise"
  device.wifi.securityMode.set="1"
  device.wifi.wpa2Ent.method="EAP-TLS"
  device.wifi.wpa2Ent.method.set="1"
  device.wifi.wpa2Ent.user="[MACAddress]"
  device.wifi.wpa2Ent.user.set="1"/>
<!--Install Certificates-->
<certificate
  device.sec.TLS.customCaCert1="[Certificate]"
  device.sec.TLS.customCaCert1.set="1"
  device.sec.TLS.profile.caCertList1="Platform1"
  device.sec.TLS.profile.caCertList1.set="1"
  device.sec.TLS.profileSelection.dot1x="PlatformProfile1"
  device.sec.TLS.profileSelection.dot1x.set="0"/>
```

How the parameters work:

device.wifi.securityMode="WPA2-Enterprise"

Establishes WPA2-Enterprise as the type of Wi-Fi security

device.wifi.wpa2Ent.method="EAP-TLS"

Establishes the WPA2-Enterprise method

device.wifi.wpa2Ent.user="[MACAddress]"

Sets the identity for the method. The MACAddress is used for the device certificate. A generic identity that is accepted by the RADIUS server is configurable by SLIC.

device.sec.TLS.customCaCert1="[certificate]"

Loads the actual certificate onto the handset. Referred to as Platform1.

device.sec.TLS.profile.caCertList1="Platform1"

Directs the handset to use the Platform2 certificate which is the customCaCert1 loaded above.

device.sec.TLS.profileSelection.dot1x

Tells the handset that for EAP-TLS authentication it will find the certificate it needs in Platform Profile 1.

Servers that use Platform certificates

Both provisioning servers and syslog servers can use certificates. If you use HTTPS or FTPS you will need to configure a certificate for the server. These are loaded and assigned exactly as described above for Wi-Fi security methods EAP-TLS or PEAP.

Spectralink recommends that you use different CA certificates for servers than you do for Wi-Fi security methods and assign the Platform Profiles accordingly. Provisioning server certificates can be configured with SLIC.

Parameter	Permitted Values	Default
device.sec.TLS.profileSelection.provisioning¹	PlatformProfile1, PlatformProfile2	Null
The TLS Platform Profile to use for provisioning, either Platform Profile 1 or TLS Platform Profile 2.		
device.sec.TLS.profileSelection.syslog¹	PlatformProfile1, PlatformProfile2	Null
The TLS Platform Profile to use for syslog, either Platform Profile 1 or TLS Platform Profile 2.		

Example: FTPS

Your configuration for implementing FTPS on a provisioning server could look like this:

```
<FTPS
device.prov.serverType="FTPS"
device.prov.serverType.set="1"
device.sec.TLS.customCaCert2="[Certificate]"
device.sec.TLS.customCaCert2.set="1"
device.sec.TLS.profile.caCertList2="Platform2"
device.sec.TLS.profile.caCertList2.set="1"
device.sec.TLS.profileSelection.provisioning="PlatformProfile2"
device.sec.TLS.profileSelection.provisioning.set="1"/>
```

How the parameters work:

device.prov.serverType="FTPS"

Sets the FTPS protocol on the provisioning server.

device.sec.TLS.customCaCert2=[certificate]

Loads the actual certificate onto the handset. Also referred to as Platform2.

device.sec.TLS.profile.caCertList2="Platform2"

Directs the handset to use the Platform2 certificate which is the customCaCert2 loaded above.

device.sec.TLS.profileSelection.provisioning="PlatformProfile2"

Tells the handset that for provisioning server authentication it will find the certificate it needs in Platform Profile 2.

#4 Common name validation

When a provisioning or syslog server sends its certificate to the client to be validated one of the default steps the phone will do is to verify that the Common Name tied to the certificate matches the name of the server it is talking to. In most cases this won't be much of a problem, but it could be an issue if you are using self-signed certificates. If the names don't match, the default behavior is to deny the connection and disconnect.

You can disable common name validation and allow the phone to ignore the name of the server and the name in the certificate. Note that this is less secure but for many environments it will be perfectly acceptable. If you need to disable common name validation use the following parameters.

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
device.sec.TLS.dot1x.strictCertCommonNameValidation If set to 1, 802.1X always verifies the server certificate for commonName/SubjectAltName match with the server hostname that the phone is trying to connect.	0 or 1	1
device.sec.TLS.prov.strictCertCommonNameValidation If set to 1, provisioning always verifies the server certificate for commonName/SubjectAltName match with the server hostname that the handset is trying to connect. Must be disabled for FTPS.	0 or 1	1
device.sec.TLS.syslog.strictCertCommonNameValidation If set to 1, syslog always verifies the server certificate for commonName/SubjectAltName match with the server hostname that the handset is trying to connect.	0 or 1	1
sec.TLS.SIP.strictCertCommonNameValidation If 1, enable common name validation for SIP.	0 or 1	1

If you are not sure whether or not you need to disable common name validation, check the logs. What you will likely see in the log of your FTPS or HTTPS server is that the device will establish a connection but will not send its authentication credentials.

Application certificates

Application certificates can be used by any application that requires a certificate that is not a Platform type. Six Application "slots" are available. Loading and assigning application certificates follows the same sequence as detailed above for Platform certificates. Application certificates are not configured by SLIC.

This section also offers two Platform certificate slots that are separate and different from the two device parameters that can be configured by SLIC.

An Application may have both its own application certificate and a corresponding custom device certificate.

Either Platform or Application slots can be used by:

- SIP
- Browser
- LDAP

#1 Loading Application certificates

Application certificates are assigned a number (1-6) and that number is used to identify the application when installing and assigning the certificate it requires. Application certificates are installed using the following parameters for application slots:

Parameter	Permitted Values	Default
sec.TLS.customCaCert.x	String	Null
The custom certificate for Application Profile x (x= 1 to 6). E.g. The certificate loaded in <code>sec.TLS.customCaCert.1</code> is referred to as <code>ApplicationProfile1</code> .		

Example of an `ApplicationProfile1` certificate:

```
sec.TLS.customCaCert.1 ="-----BEGIN CERTIFICATE-----
MIIC0DCCAjmgAwIBAgIQUCDYJiwlSU0qx54jdCZ4hDANBgkqhkiG9w0BAQUFADB6
MQswCQYDVQQGEwJVUzEUMBIGA1UEChMLU3BIY3RyYUxpbmsxDTALBgNVBAstBEVO
R1I1xJzAlBgNVBAMTHmNpc2NvdWNTOS5lbmdyLnNwZWNOcmFsaW5rLmNvbTELMAG
A1UECBMCQ08xEDAOBgNVBAcTB0JvdWxkZXIwHhcNMTI3MTkxNDE1WhcNMTcx
MTI2MTkxNDE0WjB6MQswCQYDVQQGEwJVUzEUMBIGA1UEChMLU3BIY3RyYUxpbmsx
DTALBgNVBAstBEVOR1I1xJzAlBgNVBAMTHmNpc2NvdWNTOS5lbmdyLnNwZWNOcmFsa
W5rLmNvbTELMAGAGA1UECBMCQ08xEDAOBgNVBAcTB0JvdWxkZXIwZ8wDQYJKoZI
hvcNAQEBBQADgY0AMIGJAoGBAKsCtLpHgH5ZCm/VkFbTPMe9tATQr141VwNiC31j
W7WovcQBRsfUu8nBLspfGPchtWoTfgwroVrvIHJJkzrdr3j/tQPj+9OqnnJEr6ik
HjthkMGMvFa6yvZRRCcymvSvxMLyOE2yxDalckiNviqwhRu+DhE78fqHJUldXM+
yFBVAgMBAAgVzBVMAsgA1UdDwQEAwICvDANBgNVHSUEIDAeBggrBgEFBQcDAQYI
KwYBBQUHAwIGCCsGAQUFBwMFMBOGA1UdDgQWBBQy2gOiWCJrb2ablj5W66TSGSzu
JzANBgkqhkiG9w0BAQUFAAOBgQBxDLemxeXPjloe8OZrT/f0a22lB+DbqOngByYW
/AFwvCRhqT2cb/huf4GhoADtdQ41XEzzDPu9asOJwc0cjhM39iMcXc45W+0n36l4
X7/Z9TYrWEGDfhvOMS6dV2ODYsnLm6YhOFIpl/qlrWEiCujn6rjilM2QCM6GpcFRL
zVdDoA==
-----END CERTIFICATE-----"
```

#2 Connecting the certificate to the application

In this step, you will link the certificate loaded to the defined profile. You have these options

- Platform1 or Platform2
- Applicationx (1-6) the certificates loaded above for an application.

Parameter	Permitted Values	Default
sec.TLS.profileSelection.SIP	PlatformProfile1, PlatformProfile2, ApplicationProfile1, ApplicationProfile2, ApplicationProfile3, ApplicationProfile4, ApplicationProfile5, ApplicationProfile6	PlatformProfile1
sec.TLS.profileSelection.syslog		
sec.TLS.profileSelection.LDAP		
sec.TLS.profileSelection.browser		

The Platform or Application Profile to use for the application identified in the parameter. Note that you can assign a Platform certificate to an application using this parameter.

The TLS platform profile or TLS application profile to use for applications.

Parameter	Permitted Values	Default
The TLS platform profile or TLS application profile (see preceding list) to use for the Corporate Directory.		
The TLS platform profile or TLS application profile (see preceding list) to use for SIP operations.		
The TLS platform profile to use for syslog operations.		

Example of SIP using an Application profile.

```
sec.TLS.profileSelection.SIP="ApplicationProfile1"
sec.TLS.customCaCert.1="-----BEGIN CERTIFICATE-----
MIIC0DCCAjmgAwIBAgIQUCDYJiwlSU0qx54jdCZ4hDANBgkqhkiG9w0BAQUFADB6
MQswCQYDVQQGEwJVUzEUMBIGA1UEChMLU3BIY3RyYUxpbmsxDTALBgNVBAsTBEO
R1IxJzAlBgNVBAMTHmNpc2NvdWNTOS5lbmdyLnNwZWNOcmFsaW5rLmNvbTElMAK
A1UECBMCQ08xEDAOBgNVBACTB0JvdWxkZXIwHhcNMTIxMTI3MTkxNDE1WhcNMTcx
MTI2MTkxNDE0WjB6MQswCQYDVQQGEwJVUzEUMBIGA1UEChMLU3BIY3RyYUxpbmsx
DTALBgNVBAsTBEOVR1IxJzAlBgNVBAMTHmNpc2NvdWNTOS5lbmdyLnNwZWNOcmFsa
aW5rLmNvbTElMAKGA1UECBMCQ08xEDAOBgNVBACTB0JvdWxkZXIwZ8wDQYJKoZI
hvcNAQEBBQADgY0AMIGJAoGBAKsCtLpHgH5ZCm/VkFbTPMe9tATQr141VwNiC31j
W7WovcQBR5fUu8nBLspfGPchtWoTfgwroVrvIHJkzrdr3j/tQPj+9OqnnJEr6ik
HJthkMGMvFa6yvZRRCcymvSvxMLyOE2yxDalkciNviqwhRu+DhE78fqHJUldXM+
yFBVAgMBAAGjVzBVMASGA1UdDwQEAwICvDANBgNVHSUEIDAeBggrBgEFBQcDAQYI
KwYBBQUHAWIGCCsGAQUFBwMFM80GA1UdDgQWBBQy2gOiWCJrb2ablj5W66TSGSzu
JzANBgkqhkiG9w0BAQUFAAOBgQBxDLemxeXPjloe8OZrT/f0a22lB+DbqOngByYW
/AFwvCRhqt2cb/huf4GhoADtdQ41XEzzDPu9asOJwc0cjhM39iMcXc45W+0n36l4
X7/Z9TYrWEGDfhvOMS6dV2ODYsnLm6YhOFIpl/qlrwEiCujn6rjilM2QCM6GpcfRL
zVdDoA==
-----END CERTIFICATE-----"
```

Device certificates

Device Certificates are used in the following situations:

- Mutual TLS Authentication: Allows a server to verify that a device is truly a Spectralink device (and not a malicious endpoint or software masquerading as a Spectralink device). This could be used for tasks like provisioning, or SIP signaling using TLS signaling.
- Secure HTTP (https) access to the web server on the phone at <https://<IP ADDRESS OF PHONE>>. The web server is used for certain configuration and troubleshooting activities.
- Secure communications utilizing the Spectralink Applications API.

The device can be configured to use different device certificates for each operation (or the same device certificate can be used for multiple operations). The operations available are:

- 802.1X
- Syslog
- Provisioning

- SIP
- Browser
- Presence
- LDAP

This configuration can be done:

- Using configuration files.
- From the phone menu.

There are several options for utilizing device certificates on the phone.

- A factory installed device certificate. This certificate is installed at the time of manufacture and is unique to a device (based on the MAC address) and signed by the Spectralink Certificate Authority (CA). Since it is installed at the time of manufacture, it is the easiest option for out-of-box activities; in particular, device provisioning.
- Two platform device certificates. These certificates are loaded onto the device by the system administrator and can be configured to be used for any of the following purposes: 802.1X Authentication, provisioning, syslog, SIP signaling, browser communications, presence, and LDAP.
- Six Application device certificates. These certificates are loaded onto the device by the system administrator and can be used for all of the operations listed above for platform certificates with the exception of 802.1X, syslog, and provisioning.

Configuration options are used to select which type of device certificate is used for each of the secure communication options. By default, all operations will utilize the factory installed device certificate.

To configure your web servers and/or clients to trust Spectralink factory installed device certificates, you will need to download the Spectralink Issuing CA certificate and the Root CA will need to be installed on the server. See [Appendix F: Spectralink Certificates](#) for link. You may also need to download the Intermediate CA certificates; this is determined by the authenticating server.

Platform Device Certificates

Platform device certificates can be installed using one of the following methods:

- Using a configuration file. You must enter the certificate in PEM or PKCS# certificate format.

```
device.sec.TLS.customDeviceCertX.publicCert
device.sec.TLS.customDeviceCertX.privateKey
device.sec.TLS.customDeviceCertX.set
```

where X = 1 or 2.

Assign the certificate to Platform Profile 1 or 2 using:

```
device.sec.TLS.profile.deviceCertx
```

where X = 1 or 2.

Parameter	Permitted Values	Default
device.sec.TLS.profile.deviceCert1 (TLS Platform Profile 1) device.sec.TLS.profile.deviceCert2 (TLS Platform Profile 2)	Builtin, Platform1, Platform2	Null
Choose the device certificate(s) for TLS Platform Profile 1 and TLS Platform Profile 2 to use for authentication.		
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.privateKey device.sec.TLS.customDeviceCert2.privateKey	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.privateKey</code> . All other <code><device/></code> parameters have their own corresponding <code>.set</code> parameter that will enable or disable that parameter.		

- From the phone. Navigate to **Settings > Advanced Settings > Admin Settings > TLS Security > Configure TLS Profiles > Custom Device Credentials**. You must enter a URI linking to a PEM formatted certificate in PKCS #7 certificate format.
- By generating a Certificate Signing Request (CSR). See [Generating a Certificate Signing Request](#).
- The total size of the platform certificate plus private key is restricted as follows:
 - Platform Certificate –8192 bytes.
 - Platform Private Key–4096 bytes.

If the administrator attempts to download a certificate that is too big, 'Failed to save certificate' displays on the phone's screen and a message appears in the log file (shown next).

```
0529103935|tls|4|03|Device credential invalid: Cert is not proper in the certificate
```

Application Device Certificates

Application certificates can be installed using one of the following methods:

- Using a configuration file. You must enter the certificate in PKCS #7 certificate format. The configuration parameters are:

```
sec.TLS.customDeviceCert.x
sec.TLS.customDeviceKey.x
```

Assign the certificate to a Platform or Application Profile using:

```
sec.TLS.profile.x.deviceCert
```

where x = 1-6.

Parameter	Permitted Values	Default
sec.TLS.customDeviceCert.x	String	Null
The custom device certificate for TLS Application Profile x (x= 1 to 6).		
sec.TLS.customDeviceKey.x	String	Null
The custom device certificate private key for TLS Application Profile x (x= 1 to 6).		
sec.TLS.profile.x.deviceCert	Factory, Platform1, Platform2, Application1, Application2, Application3, Application4, Application5, Application6	Factory
The device certificate to use for TLS Application Profile x (x = 1 to 6).		



Caution: Exposing the device key is insecure

This method is available but exposes the private key. This is not secure and is generally not recommended. Allow the phone to generate a certificate signing request instead.

Example custom device Cert and Key

```
sec.TLS.customDeviceCert.1="MIICIDCCAYmgAwIBAgIDASNFMAGCSqGSIb3DQEBAUAMFUCzAJBgNVBAYTAIVT
MQswCQYDVQQLIEwJDTZEqMA4GA1UEBxMHQm91bGRlcjESMBAGA1UEChMJVGhpc1N0dWZmMRMwEQYDVQDEwp
UaG9zZUR1ZGVzMBA4XDTEyMDYxMzIxMzIzMFoXDTEzMDYxMzIxMzIzMFowVTElMAkGA1UEBhMCVVMxMzAJBgNVBA
gTAKNPMRAwDgYDVQQHEwdCb3VsZGVyMRlwEAYDVQQKEwluUaGlzU3R1ZmYxEzARBgNVBAMTCiRob3NIRHVkZXM
wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBA0JNx7uP/fsY6RFQd4F97nKdl8svtSNUSzpAqGv5/dEIRRRmGrcaEcG
WJZ1SnsyYVlbp9qmMAaHe9caTlIT9jKJ7LFfURQgI57G4TKxutxArX5dPnuNuL1xPvhZMOMhiuJLfNb1JtOasecIRGZ2B5L
kHAZjUgMLfxP6HePrQKtAgMBAAEwDQYJKoZIhvcNAQEEBQADgYEAf+r1H1TErx7A9slvLgVKyvbDOYcotH5DetT7F//
hWp4blcDDuQC8w2oj9WpyCGCvPkYERw3yv7Un7Qgra6CtWF59cmrvk9UCzmkCO0XrqRNT9V2MNPd4KkrD2o8S7B0
l8162p9D3D5JHXINaUekICh1ziO3Qp54HTJ4n4iOonU="

sec.TLS.customDeviceKey.1="MIICdgIBADANBgkqhkiG9w0BAQEFAASCAMAwggJcAgEAAoGBA0JNx7uP/fsY6RFQd4
F97nKdl8svtSNUSzpAqGv5/dEIRRRmGrcaEcGWJZ1SnsyYVlbp9qmMAaHe9caTlIT9jKJ7LFfURQgI57G4TKxutxArX5dPn
uNuL1xPvhZMOMhiuJLfNb1JtOasecIRGZ2B5LkHAZjUgMLfxP6HePrQKtAgMBAAECgYA6jFYpA6OaXL9wzU3g0S0Awk
WEojDp+68XrV8uGKcOxQeqg6tAVVwV1Y8MK9UoGz9W1sxkA4P9k7c4A7IRGdseHxR6LeHzA9Dt/7p8Kz+isw+ac/M
EFwI5wfrWM8HVwrZ5ZVZ5qV2q4EC3zC2DDwRyqRII0+8QTjnelEYrhlaWQQJBAPWMMktq058sMhRkdbdplpBtnglwUG9
DmV536umM8jvoU8DtrQ6zChAoiwmHAZfeSqaag0sWW4aPLU4zrpoVickCQQDytrXySEBQcnBA1u2CcZMJgS9m2tTg
w3CCILqWDVLkbhGkDL/x4wE307+37hkWrPx9FW0BjDFCNVVLgMli56PFAk81BVB+0rVx+T0QrOARojj4ekSfIYHAJ9V
WaVRksNqu4dU2c78d2UHWfsSuk2PE/hbsV0zU/x21XgzlZScD9dnJAKB+GWEkNFd9tEcP2Npi2CD9Yim/dVf+Qmv02
9Ko3NlBJtJxJedJLmJXsbE4CAGj2d/9h+Ty9/vFGRbE/3fRDN3fFAkEAz2P/ZtUzjU2qG48wUumV3DzGpTRaaLqVqnwjy
4Cey2dWTXClkf9s7fwhntSGiZZRtbgnok+A/8w/ry4tLTIA=="
```

- From the phone. Navigate to **Settings > Advanced > Admin Settings > TLS Security > Configure TLS Profiles > Custom Device Credentials**. You must enter a URI linking to a PEM formatted certificate in PKCS #7 certificate format.
- By generating a Certificate Signing Request (CSR). See below. There is no size constraint on the application certificate and private key.

Cipher Suite Parameters

The handset administrator can control which cipher suites will be offered/accepted during TLS session negotiation. The handset supports the cipher suites shown in the next table. The 'Null

Cipher' is a special case option which will not encrypt the signaling traffic, and is useful for troubleshooting purposes.

You will only need to configure cipher suite parameters if you are not using the default.

<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

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
device.sec.TLS.profile.cipherSuite1 (Platform Profile 1) device.sec.TLS.profile.cipherSuite2 (Platform Profile 2)	String	Null
The cipher suites to use for Platform Profile 1 and Platform Profile 2) Use the Cipher Suites table above.		
device.sec.TLS.profile.cipherSuiteDefault1 (Platform Profile 1) device.sec.TLS.profile.cipherSuiteDefault2 (Platform Profile 2)	0 or 1	Null
The cipher suite to use for Platform Profile 1 and 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.		
sec.TLS.cipherList	String	"RSA:!EXP:!LOW:!NULL:!MD5:@STRENGTH"
The global cipher list parameter.		
sec.TLS.browser.cipherList	String	NoCipher
The cipher list for the browser.		
sec.TLS.LDAP.cipherList	String	NoCipher
The cipher list for the corporate directory.		
sec.TLS.prov.cipherList	String	NoCipher
The cipher list for provisioning.		
sec.TLS.SIP.cipherList	String	NoCipher
The cipher list for SIP.		
sec.TLS.syslog.cipherList	String	NoCipher
The cipher list for syslog.		

Generating a Certificate Signing Request

You may need a certificate to perform a number of tasks, for example, TLS with mutual authentication. For version 4.12 and above, the CSR is 2048 bits, while 4.11 and below generated 512 bit keys.

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 handset:

- Using a configuration file
- Through the handset's user interface

To generate a certificate signing request on a Spectralink handset:

- 1 Navigate to **Settings> Advanced Settings> [enter password]> Administration Settings> Generate CSR**.
- 2 From the Generate CSR Screen, enter information as shown next. You must fill in the Common Name and Country fields. The Organization, Email Address, Country, and State fields are optional.

- 3 Press Generate.
A message CSR generation completed displays on the handset's screen.

Downloading Certificates to a Spectralink Phone

You can download certificates to a Spectralink handset by specifying a URL where the certificate is currently stored. You can install up to eight CA certificates and eight device certificates on the handset. You can refresh certificates when they expire or are revoked. You can delete any CA certificate or device certificate that you install.



Maximum Size for Certificates

The maximum certificate size on both Platform CA1 and Platform CA2 is 4KB.

To download a certificate to a Spectralink handset:

- 1 Navigate to **Settings> Advanced Settings> [enter password]> Administration Settings> TLS Security** and select Custom CA Certificates or Custom Device Certificates.
- 2 Select one of the certificate slots from the displayed list, then press the **Install** softkey. When prompted, enter the administrative password and press the **Enter** softkey. The default administrative password is **456**.
- 3 Enter the URL where the certificate is stored.
For example, <http://bootserver1.vancouver.Spectralink.com/ca.crt>
- 4 Select the **Enter** softkey.
The certificate is downloaded. The certificate's MD5 fingerprint displays to verify that the correct certificate is to be installed.
- 5 Select the **Accept** softkey.
The certificate is installed successfully.
The appropriate certificate menu displays the certificate's common name.

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. DNS record lookup behavior may additionally be influenced if the transport mechanism or port number are specified in the phone's configuration. If the phone is allowed to operate using the default configuration parameter combination of transport method and port number, the handset will attempt record lookups in the following order until it receives a successful lookup response; NAPTR, then SRV, then A record.

If a transport method is specified to be something other than DNS NAPTR, the phone will skip the attempt to query for a DNS NAPTR record, and will instead attempt a SRV record lookup for the proxy/registrar address using the transport mechanism specified, followed by an A name record lookup if the SRV record lookup fails. If a port number is specified via the configuration

parameters, the phone will not attempt NAPTR or SRV record lookups. The only lookup will be an A record. If no port is specified via configuration parameters, and none is found through DNS mechanisms, port 5060 will be used by default for TCP or UDP transport types, and 5061 will be used as the port number for TLS. 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 handset will go through the following steps to connect the call:

- 1 The handset will try to call the working server.
- 2 If the working server does not respond correctly to the INVITE, the handset 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 handset 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.

**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 handset 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 handset call proceeds using the working server. To prevent this issue, long TTLs should be used. Spectralink recommends deploying an on-site DNS server as part of the redundancy solution.

Phone Configuration

The handsets at the customer site are configured as follows:

- Server 1 (the primary server) will be configured with the DNS name 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 handset for more than two servers per registration, but you need to exercise caution when doing this to ensure that the handset and network load generated by registration refresh of multiple registrations does not become excessive. This would be of particular concern if a handset 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 handset 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 handset 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.



Failover to Servers that are Not Registered

If `reg.x.server.y.register` is set to 0, the handset will not register to that server. However, an INVITE will fail over to that server if all higher priority servers are down.

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

Summary

Parameter	Used to:
<code>volpProt.SIP.requestValidation.x.method</code>	Specify what type of validation to perform
<code>volpProt.SIP.requestValidation.x.request</code>	Set the name of the method for which validation will be applied
<code>volpProt.SIP.requestValidation.x.request.y.event</code>	Determine which events within the Event header should be validated

Table 10-2: Incoming Signal Validation

Parameter	Permitted Values	Default
<code>volpProt.SIP.requestValidation.x.method</code>¹	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>		
<code>volpProt.SIP.requestValidation.x.request</code>¹	Null, INVITE, ACK , BYE, REGISTER, CANCEL, OPTIONS, INFO, MESSAGE, SUBSCRIBE, NOTIFY, REFER, PRACK, UPDATE	Null
<p>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.</p>		
<code>volpProt.SIP.requestValidation.x.request.y.event</code>¹	A valid string	Null
<p>Determines which events specified with the Event header should be validated; only applicable when <code>voIpProt.SIP.requestValidation.x.request</code> is set to SUBSCRIBE or NOTIFY. If set to Null, all events will be validated.</p>		

¹ Change causes handset to restart or reboot.

Instant Messaging

You can use Microsoft Office Communications Server (OCS) 2007 R2 and Microsoft Lync servers for instant messaging.

In our IM example, line 2 becomes the IM line. Lync2010 (or 2013, depending on your software version) is deployed. For User Profiles Deployment, the reg.2 parameters are located in the login.cfg template.

Table 10-3: IM

Parameter	Permitted Values	Default
feature.messaging.enabled	0 or 1	0
If 0, the instant messaging feature is disabled. If 1, the feature is enabled.		
feature.presence.enabled	0 or 1	0
If 0, the presence feature — including user status — is disabled. If 1, the presence feature is enabled with the status options.		
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 2013 or 2010. If 1, telephony calls are enabled on this registration.		
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.server.y.address	dotted-decimal IP address or hostname	Null
The IP address or host name of a SIP server that accepts registrations. If not Null, all of the parameters in this table will overrule the parameters specified in <code>voIpProt.server.*</code> . <i>Note:</i> If this parameter is set, it will overrule even if the DHCP server is available. If this registration is used for Microsoft Office Communications Server 2007 R2, this parameter must be in the form <code>OCShostname.OSCdomain_name</code> . For IM and Presence, use <code>reg.2.server.1.address</code> to contact and register with the Lync server.		
reg.x.server.y.port	0, 1 to 65535	0
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 the outbound proxy can route calls without the handset being registered to it, set this value to 0.		
reg.x.server.y.specialInterop	standard, ocs2007r2, lcs2005, lync2010, lync2013	standard
Specify if this registration should support Microsoft Office Communications Server 2007 R2 (ocs2007r2), Microsoft Live Communications Server 2005 (lcs2005), or Microsoft Lync 2010 or 2013 (lync2010 or lync2013). <i>Note:</i> To use instant messaging, set this parameter to ocs2007r2.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
reg.x.server.y.transport	Null, DNSNaptr, TCPpreferred, UDPOnly, TLS, TCPOnly	DNSNaptr
<p>The transport method the handset uses to communicate with the SIP server.</p> <p>Null or DNSNaptr – if reg.x.server.y.address is a hostname and reg.x.server.y.port is 0, do NAPTR then SRV look-ups to try to discover the transport, ports and servers, as per RFC 3263. If reg.x.server.y.address is an IP address, or a port is given, then UDP is used.</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>		
roaming_buddies.reg	Null, 1 to 34	Null
<p>The index of the registration which has roaming buddies support enabled. If Null, the roaming buddies feature is disabled.</p> <p>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.</p>		
sec.TLS.customCaCert.x	String	Null
<p>The custom certificate for TLS Application Profile x (x= 1 to 6). This parameter is not in the template but may need to be added. It is not required if you use DHCP option 43 to tell the handset where to get its certificate automatically but if that is not available , then add the certificate using this parameter.</p>		
sec.TLS.profileSelection.SIP	a TLS profile	PlatformProfile1
<p>The TLS platform profile or TLS application profile to use for SIP operations. Permitted values are:</p> <ul style="list-style-type: none"> PlatformProfile1 PlatformProfile2 ApplicationProfile1 ApplicationProfile2 ApplicationProfile3 ApplicationProfile4 ApplicationProfile5 ApplicationProfile6 		
volpProt.SIP.mtls.enable	0 or 1	1
<p>If 0, TLS with mutual authentication is disabled. If 1, TLS with mutual authentication is enabled. Used in conjunction with Microsoft Lync 2013 or 2010.</p>		
volpProt.SIP.IM.autoAnswerDelay	0 to 40, seconds	10
<p>The time interval from receipt of the instant message invitation to automatically accepting the invitation. If users have a PC that is logged to their IM account, should the PC auto-answer incoming IMs if no action is taken on the handset? If yes, set to 30, if no, set to 10.</p> <p>Yes: volpProt.SIP.IM.autoAnswerDelay="30"</p> <p>No: volpProt.SIP.IM.autoAnswerDelay="10"</p>		

Table 10-4: Setting Up Microsoft Office Communications Server 2007 R2 Integration

<i>Parameter</i>	<i>Used to:</i>
feature.presence.enabled	Turn the presence feature on or off
feature.messaging.enabled	Turn the messaging feature on or off

Parameter	Used to:
reg.x.lineKeys	Specify the number of line keys to use for a single registration

Parameter	Permitted Values	Default
feature.presence.enabled¹	0 or 1	0
If 0, the presence feature — including user status — is disabled. If 1, the presence feature is enabled with status options.		
feature.messaging.enabled	0 or 1	0
If 0, the instant messaging feature is disabled. If 1, the feature is enabled.		
reg.x.lineKeys	1 to 24	1
Specify the number of line keys to use for a single registration.		

¹ Change causes handset to restart or reboot.

Example Instant Messaging Configuration

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

The screenshot displays the Polycom Config tool interface. On the left, the 'Tree View' shows the configuration hierarchy: 'PolycomConfig' > 'im'. The 'im' folder is expanded, showing parameters such as 'feature.messaging.enabled', 'feature.presence.enabled', 'reg.2.telephony', 'reg.2.auth.useLoginCredentials', 'reg.2.server.1.specialInterop', 'reg.2.server.1.transport', 'roaming_buddies.reg', 'sec.TLS_profileSelection.SIP', 'voIpProt.SIP.mtls.enable', and 'voIpProt.SIP.IM.autoAnswerDelay'. On the right, the 'XSL Output' pane shows the XML configuration for these parameters, including the 'version' attribute, the 'polycomConfig.xsd' schema, and the 'site.cfg template for FLAT DEPLOYMENT'. The output shows the values for the parameters, such as '1' for 'feature.messaging.enabled' and 'feature.presence.enabled', '0' for 'reg.2.telephony', '1' for 'reg.2.auth.useLoginCredentials', 'lync2010' for 'reg.2.server.1.specialInterop', 'TLS' for 'reg.2.server.1.transport', '2' for 'roaming_buddies.reg', 'ApplicationProfile1' for 'sec.TLS_profileSelection.SIP', '0' for 'voIpProt.SIP.mtls.enable', and '40' for 'voIpProt.SIP.IM.autoAnswerDelay'.

Table 10-5: Other Instant Messaging Parameters

Parameter	Permitted Values	Default
messaging.maxImMessages	10 to 1000	1000
The maximum number of instant messages allowed.		
messaging.quickNotes.x	String of up to 128 characters	Null
Up to 10 (x =1 to 10) quick notes for use in instant messages		

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 the next table 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 10-6: IP Type-of-Service (ToS)

Parameter	Permitted Values	Default
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	Null, 0 to 63	Null
Specify the DSCP of packets. If the value is not null, this parameter will overrule the other <code>qos.ip.rtp.*</code> parameters. The default value is Null, so the other <code>qos.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.		

¹ Change causes handset to restart or reboot.

<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



Troubleshooting: Trouble with WMM-AC?

Phones deployed prior to software release 4.3 use different defaults for the three Ethernet parameters listed below. If you have trouble with connectivity when WMM-AC is deployed, configure these three parameters with the defaults as shown. You can find them in the `everything.cfg` file.

Table 10-7: Quality of Service (Type-of-Service) Parameters

Parameter	Permitted Values	Default
qos.ip.callControl.dscp¹	Null, 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 overrule the other qos.ip.callControl.* parameters. The default value is Null, so the other qos.ip.callControl.* parameters will be used if no value is entered.		
qos.ethernet.rtp.user_priority	0-7	6
Real-Time Protocol (RTP) packets.		
qos.ethernet.callControl.user_priority	0-7	4
User-priority for call control packets.		
qos.ethernet.other.user_priority	0-7	0
User-priority for packets that do not have a per-protocol setting.		

¹ Change causes handset to restart or reboot.

Logging Parameters

Logs formed by these logging parameters are written to the provisioning server. A small subset of these parameters can be set for a single phone in the Admin menu at **Advanced Settings> Administrative Settings> Logging**. (Lync only.)



Caution: Changing the Logging Parameters

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

The event logging system supports the following classes of events:

Table 10-8: Logging Levels

Logging Level	Interpretation
0	Debug only
1	High detail event class
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 10-9: Event Timestamp Formats

Type	Example
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/>`

Table 10-10: Logging Level, Change, and Render Parameters

Parameter	Permitted Values	Default
log.level.change.xxx	0 to 6	4
Controls the logging detail level for individual modules. These are the input filters into the internal memory-based log system which produces the [MACaddress]-app.log file. The most commonly-used values for xxx are listed below. Additional log modules can be found in the log.cfg file.		
app1	push	
brow	sip	
cfg	so	
dot1x	ticket	
httpd	tls	
ice	utilm	
key	wlan	
pps	wmgr	
This parameter and the associated render level (see below) can be set for a single phone in the Admin menu at Advanced Settings> Administrative Settings> Logging. (Lync only.)		
log.render.file	0 or 1	1
Set to 1. Spectralink 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 handset will upload all logs that have not yet been uploaded, and erase half of the logs on the handset. The administrator may use Web browser to read all logs on the handset.		
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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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	86400
Time in seconds between log file uploads to the provisioning server. The default is 86400 which is 24 hours. Note: The log file will not be uploaded if no new events have been logged since the last upload.		
log.render.level	0 to 6	1
Specifies the lowest class of event that will be rendered to the [MACaddress]-app.log files. The log.render.level maps to severity as follows: 0 -> Debugging 1 -> High Detail 2 -> Moderate Detail 3 -> Low Detail 4 -> Minor Error 5 -> Major Error 6 -> Fatal Error		
log.render.realtime	0 or 1	1
Set to 1. Spectralink recommends that you do not change this value.		
log.render.stdout	0 or 1	0
Set to 1. Spectralink recommends that you do not change this value. Note that on Spectralink handsets, the default value is 0.		
log.render.type	0 to 2	2
Refer to Table 10-9: Event Timestamp Formats for timestamp type.		

<sched/>

The handset can be configured to schedule certain advanced logging tasks on a periodic basis. These parameters should be set in consultation with Spectralink 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 10-11: Logging Schedule Parameters

<i>Parameter</i>	<i>Permitted Values</i>
log.sched.x.level	0 to 5, default 3
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.	
log.sched.x.name	alphanumeric string
Name of an internal system command to be periodically executed. To be supplied by Spectralink.	
log.sched.x.period	positive integer, default 15
Seconds between each command execution. 0=run once	

<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> .	

Microsoft Lync Server 2013/2010 Integration

See the Microsoft interoperability Guides for Microsoft Lync Server 2013 and Microsoft Lync Server 2010 for complete information.

Network Address Translation (NAT)

The handset 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 handset'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 handset's behalf on a per-handset basis. Note that the source port in transmitted packets is the same as the associated listening port used to receive packets.

These parameters define port and IP address changes used in NAT traversal. The port changes will change the port used by the handset, 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 10-12: 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 for handsets to 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 handset will not send out keep-alive messages. The Microsoft Live Communications Server 2005 keep-alive feature will overrule this parameter. If you want to deploy handsets 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 handsets 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> .		

Parameter	Permitted Values	Default
nat.signalPort¹	1024 to 65535	0
The port used for SIP signaling. Overrides <code>voIpProt.local.port</code> .		

¹ Change causes handset to restart or reboot.

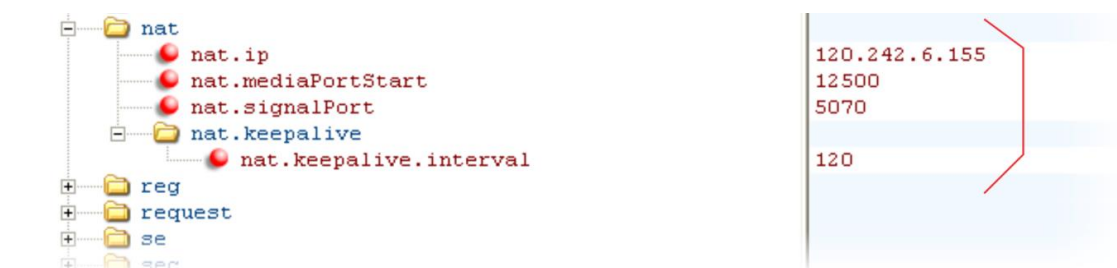
Example Network Address Translation Configuration

The parameter `nat.ip` is the public IP that you want to advertise in SIP signaling. The 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 overrule the value set in `tcpIpApp.port.rtp.mediaPortRangeStart`. In the example below, the starting port is 12500 and the handset will cycle through start-port +.

The parameter `nat.signalPort` specifies the port that the handset will use for SIP signaling. This parameter will overrule `voIpProt.local.Port`. In the example below, the handset 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 handsets 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 handset will send the keepalive every 120 seconds.



Provisioning Server System Settings

This parameter's settings control aspects of the handset's provisioning server system.

Table 10-13: Provisioning Parameters

Parameter	Permitted Values	Default
prov.configUploadPath	string	Null
The directory - relative to the provisioning server - where the handset uploads the current configuration file when the user selects Upload Configuration. If set to Null, use the provisioning server directory.		

¹ Change causes handset to restart or reboot.

<request/>

These settings control the handset's behavior when a request for restart or reconfiguration is received.

Table 10-14: Configuration Request Parameter

Parameter	Permitted Values	Default
request.delay.type¹	audio, call	call

Specify when the handset 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 handset — regardless of the call state. If set to **call**, the request should be executed once there are no calls —in any state — on the handset.

¹ Change causes handset to restart or reboot.

Security <sec/>

These parameters affects the security features of the handset.

Table 10-15: General Security Parameters

Parameter	Permitted Values	Default
sec.tagSerialNo¹	0 or 1	0

If 0, the handset does not advertise its serial number (MAC address) through protocol signaling. If 1, the handset may advertise its serial number through protocol signaling.

¹ Change causes handset to restart or reboot.

<srtp/>

As per RFC 3711, you cannot turn off authentication of RTCP.

Table 10-16: SRTP Parameters

Parameter	Permitted values	Defaults
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.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.lifetime¹	Null, 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 or Null, the master key lifetime is not set. If set to a valid value (at least 1024, or a power such as 2¹⁰), 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. *Note:* Setting this parameter to a non-zero value may affect the performance of the handset.

<i>Parameter</i>	<i>Permitted values</i>	<i>Defaults</i>
sec.srtp.mki.enabled¹	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.length¹	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.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.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 Change causes handset to restart or reboot.

<dot1x><eapollogoff/>

Table 10-17: 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 handset will not send an EAPOL Logoff message on behalf of the disconnected supplicant. If 1, the feature is enabled and the handset will send an EAPOL Logoff message on behalf of the disconnected supplicant connected to the handset's secondary (PC) port.		
sec.dot1x.eapollogoff.lanlinkreset¹	0 or 1	0
If 0, the handset 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 handset to restart or reboot.

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 10-18: Secure Real Time Transport Protocol](#). When this feature is enabled, handsets 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 handsets set up SRTP for a call, see [RFC 4568](#).

Authentication proves to the handset receiving the RTP/RTCP streams 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 handset's processor usage.

Summary

<i>Parameter</i>	<i>Used to:</i>
sec.srtp.enable	Enable SRTP
sec.srtp.offer	Include secure media in SDP of SIP INVITE
sec.srtp.offer.*	Include crypto in offered SDP
sec.srtp.require	Secure media stream required in all SIP INVITEs
sec.srtp.requireMatchingTag	Check tag in crypto parameter in SDP
sec.srtp.sessionParams.*	Specify if the handset offers and/or requires: RTP encryption, RTP authentication, and RTCP encryption

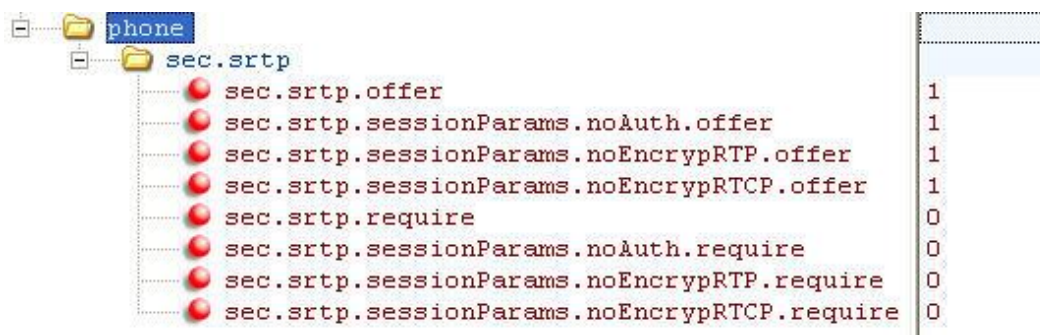
Table 10-18: Secure Real Time Transport Protocol

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
sec.srtp.enable¹	0 or 1	1
If 0, the handset always declines SRTP offers. If 1, the handset accepts SRTP offers.		
sec.srtp.offer¹	0 or 1	0
If 1, the handset 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 handset initiating (offering) a phone call. If 0, no secure media stream is included in SDP of a SIP invite.		
sec.srtp.offer.HMAC_SHA1_32¹	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_80¹	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.		

Parameter	Permitted Values	Default
sec.srtp.require¹	0 or 1	0
If 0, secure media streams are not required. If 1, the handset 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.requireMatchingTag¹	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.sessionParams.noAuth.offer¹	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 <code>UNAUTHENTICATED_SRTP</code> session parameter is sent when initiating a call.		
sec.srtp.sessionParams.noAuth.require¹	0 or 1	0
If 0, authentication of RTP is required. If 1, no authentication of RTP is required; a call placed to a handset configured with this parameter must offer the <code>UNAUTHENTICATED_SRTP</code> 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.noEncrypRTCP.offer¹	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 <code>UNENCRYPTED_SRTP</code> session parameter is sent when initiating a call.		
sec.srtp.sessionParams.noEncrypRTCP.require¹	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 handset configured with <code>noAuth.require</code> must offer the <code>UNENCRYPTED_SRTP</code> session parameter in its SDP. If this parameter is set to 1, <code>sec.srtp.sessionParams.noEncrypRTCP.offer</code> will also be set to 1, regardless of the value in the configuration file.		
sec.srtp.sessionParams.noEncrypRTP.offer¹	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 <code>UNENCRYPTED_SRTP</code> session parameter is sent when initiating a call.		
sec.srtp.sessionParams.noEncrypRTP.require¹	0 or 1	0
If 0, encryption of RTP is required. If 1, no encryption of RTP is required. A call placed to a handset configured with <code>noAuth.require</code> must offer the <code>UNENCRYPTED_SRTP</code> session parameter in its SDP. If set to 1, <code>sec.srtp.sessionParams.noEncrypRTP.offer</code> will also be set to 1, regardless of the value in the configuration file.		

¹ Change causes handset to restart or reboot.

In Example 1, the `srtp_1.cfg` configuration file is shown below:

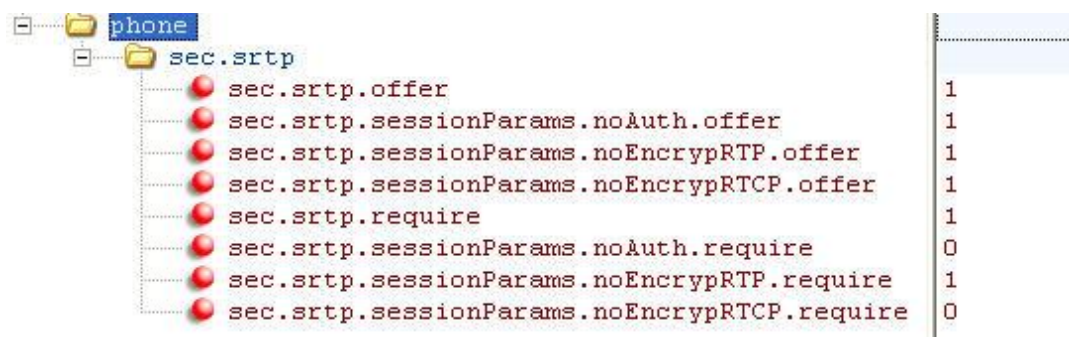


This would result in an offer (SIP INVITE with SDP) with 8 crypto attributes with the following session parameters:

```
<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 handset receiving this call should choose the most secure crypto it can support based on the SRTP “require” settings 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:



<code>sec.srtp.offer</code>	1
<code>sec.srtp.sessionParams.noAuth.offer</code>	1
<code>sec.srtp.sessionParams.noEncrypRTP.offer</code>	1
<code>sec.srtp.sessionParams.noEncrypRTCP.offer</code>	1
<code>sec.srtp.require</code>	1
<code>sec.srtp.sessionParams.noAuth.require</code>	0
<code>sec.srtp.sessionParams.noEncrypRTP.require</code>	1
<code>sec.srtp.sessionParams.noEncrypRTCP.require</code>	0

This would result in an offer (SIP INVITE with SDP) with 4 crypto attributes with the following session parameters:

```
UNENCRYPTED_S RTP 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, every crypto includes the `UNENCRYPTED_S RTP` session parameter because it is required.

If nothing compatible is offered based on the receiving handset’s STRP “require” settings, then the call is rejected or dropped.

Server Redundancy

Server redundancy is often required in VoIP deployments to ensure continuity of phone service if, for example, the call server needs to be taken offline for maintenance, the server fails, or the connection between the handset and the primary server fails.

Terminology

Before you read this section, take a moment to familiarize yourself with the following definitions:

Primary server The primary server is the highest priority server in a group of servers with an active registration. All communications route to the primary server first, unless the handset environment is configured otherwise.

Secondary server A secondary server backs up a primary server when the primary server fails. A secondary server may offer the same, or lesser, functionality than the primary server.

Server redundancy This refers to the practice of employing multiple servers so that when a primary server fails, a secondary server can take over.

Failover A type of server redundancy in which a secondary server takes over all the functions of the primary server when the primary server fails. No handset functionality is lost when the secondary server takes over.

Fallback In this mode, a second call server of lesser capability (such as a 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, voicemail, presence, and Message Waiting Indicator). Spectralink 84-Series handsets support configuration of multiple servers per SIP registration for this purpose.

Re-registration on failover A redundancy requirement in which a handset must successfully register with a server before communications can take place. If a server fails and a handset must communicate with another server (for example, a secondary server), the handset must register with the secondary server before communications can take place.

Failback A type of server redundancy in which a secondary server remains operational while communications with a primary server are retried to see if the primary server is functioning again. In certain configurations, the handset attempts to re-register with the primary server during fallback.

Register transaction A register transaction associates a handset with a particular location, such as an IP address. A handset sends a message—called a 'REGISTER' message—informing the server of its location.

Registrar or Registrar server A registrar server accepts registrations, or location information, from handsets and places this information in a database. Every handset must register its current location with a Registrar server before the handset can communicate with a server.

About the Optional Failover Behaviors

Spectralink handsets rely on two server redundancy technologies: failover and fallback. Using these technologies, multiple servers are set up so that when the primary server fails, a secondary server can take over. In some cases, a combination of the two may be deployed. Consult your SIP server provider for recommended methods of configuring handsets and servers for failover configuration.

Re-register on failover only applies to servers that use the failover method.

With failover, all servers (primary and secondary) share the same registration data. In this scenario, secondary servers support all the features that the primary server supports.

With re-register on failover disabled—the default behavior—when a handset's registration request is diverted to a secondary server, the handset doesn't have to register with the secondary server.

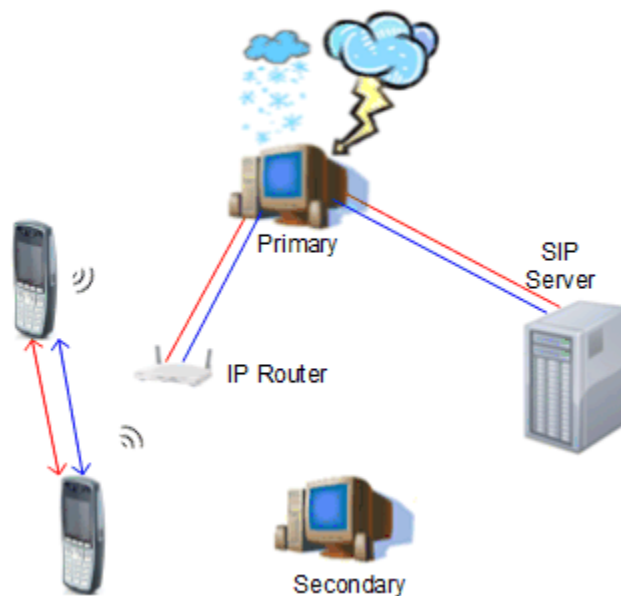
A potential issue with the default failover behavior is that some servers or intermediate SIP-aware devices may limit a handset's functionality if the server hasn't successfully processed the handset's registration request. With re-register on failover enabled, when a handset's SIP request is diverted to a secondary server, the handset will first register with the secondary server.

Optional behaviors enhance redundancy features. These behaviors include re-registration and recovery behaviors, as well as a behavior that controls how existing calls—calls that are established before a server fails—are treated.

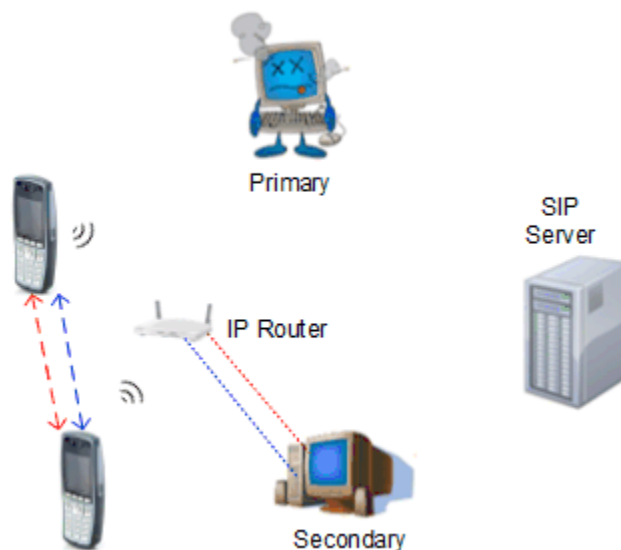
- **Re-registration behavior** The handset must complete a new registration with the failover server before communications can take place between the handset and the failover server.
- **Recovery behavior** This behavior requires handsets to communicate with the server that processed the last successful transaction, rather than always with the primary server. If this behavior is configured, you must set up rules to determine when the primary server is tried again (for example, whenever the handset has a new request, or after a specific period of time). The secondary server will remain operational while the handset is trying to re-register with the primary server ('failback').
- **Behavior for existing calls** This behavior controls the handling of calls established through the failed server after failover occurs. When this behavior is enabled, handsets won't communicate with failed servers that recover until failback succeeds. This helps avoid situations in which large numbers of handsets toggle rapidly between servers when there is an intermittent failure.

The following diagrams show how a network uses the re-registration on failover behavior. In the diagrams, primary and secondary re-registration on failover-aware Session Border Controllers (RRoFO-aware SBCs) are set up so that if a handset can't communicate with the primary RRoFO-aware SBC, the handset can attempt to register, and then communicate, with a secondary RRoFO-aware SBC.

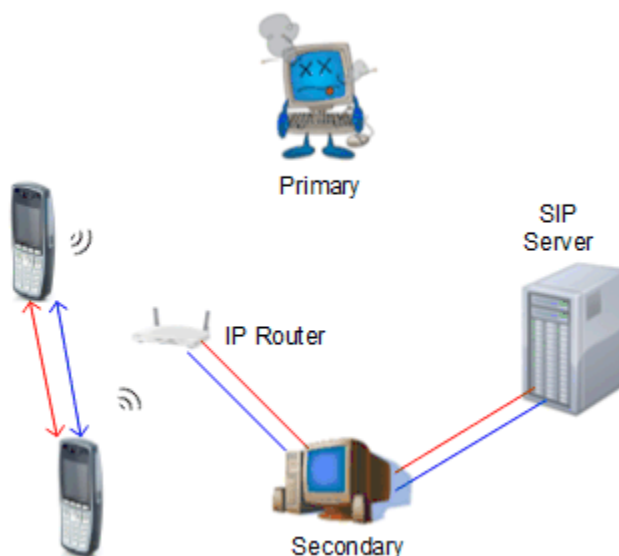
In the following diagram, handsets are communicating with a primary RRoFO-aware SBC that is just about to fail.



When the primary RRoFO-aware SBC fails, handsets can no longer communicate with it. The handsets will attempt to register with a secondary RRoFO-aware SBC, as shown next.



If registration is successful (as shown next), handsets can communicate with the secondary RRoFO-aware SBC, and traffic flow will continue without interruption.



Enabling Proxy Servers

If your implementation uses proxy servers, make sure you enable the use of proxy servers in your configuration file(s). If you've defined reg parameters, set reg.x.server.y.useOutboundProxy to 1.



Note: Compatibility with Microsoft Lync

The concurrent failover/fallback feature is not compatible with Microsoft Lync.

Fallback Deployments

Fallback deployments are most common in satellite office situations where the primary server is in a distant location and the WAN connection may fail. The basic configuration uses a reg.1.server.1.address and reg.1.server.2 address and the handset registers to both servers. When a handset makes a call the invites are sent to server.1. When server.1 is unavailable, the calling handset detects this condition and sends the invite to server.2. The handset continues to attempt to register to server.1 with every call. When server.1 comes back online, invites are once again routed to it.

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.
- 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.

Failover Deployments

Failover deployments rely on a SRV (Service Record) built in the DNS server. Phones then use the domain name as the reg.1.server.1 address, and use the NAPTR (Name Pointer) method (the default setting in the 84-Series) for querying the DNS server. The domain contains the SRV records which provide a list of potential hosts and connection protocols (TCP, UDP, or TLS), ordered in the sequence you wish clients to attempt to connect.

Most configuration is done in the DNS server, not in the handset. The basic configuration uses just one registration (reg.1).

DNS Server Unavailability

Failover redundancy can only be utilized when the configured IP server hostname resolves (through SRV or A record) to multiple IP addresses. Some configurations are not able to take advantage of failover redundancy due to non-availability of the DNS server. In these situations, the handsets support the ability to statically configure a set of DNS NAPTR SRV and/or A records into the handset. See Static DNS Cache in the Special Use Cases Chapter.

Redundancy Parameters

Summary

Parameter	Used to:
reg.x.server.y.failOver.*	Specify server redundancy options including failback mode, failback timeout, and failover registration behaviour
reg.x.auth.optimizedInFailover	Specify which server to contact if failover occurs
reg.x.outboundProxy.failOver.*	Override the default server redundancy options for a specific registration

The following table shows the parameters you need to define.

Table 10-19: Basic FailOver Configuration File Settings

Parameter	Permitted Values	Default
reg.x.server.y.failOver.reRegisterOn If you use proxy servers, set this parameter instead: reg.x.outboundProxy.failOver.reRegisterOn When set to 1, the handset 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 handset won't attempt to register with the secondary server, since the handset will assume that the primary and secondary servers share registration information. <i>Note: When this parameter is enabled, the authOptimizedInFailover parameter is automatically enabled.</i>	0 or 1	0
reg.x.server.y.failOver.failRegistrationOn If you use proxy servers, set this parameter instead: reg.x.outboundProxy.failOver.failRegistrationOn When set to 1, and the reRegisterOn parameter is enabled, the handset will silently invalidate an existing registration (if it exists), at the point of failing over.	0 or 1	0

Parameter	Permitted Values	Default
When set to 0, and the reRegisterOn parameter is enabled, existing registrations will remain active. This means that the handset will attempt failback without first attempting to register with the primary server to determine if it has recovered.		
reg.x.server.y.failOver.onlySignalWithRegistered If you use proxy servers, set this parameter instead: reg.x.outboundProxy.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 handset 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). Note: This setting primarily affects signaling associated with existing dialogs that are RFC-mandated to communicate with the servers through which they were established. A new dialog's signaling will be sent through the 'current' server.		
reg.x.server.y.failOver.failBack.mode If you use proxy servers, set this parameter instead: reg.x.outboundProxy.failOver.failBack.mode	DNSTTL, registration, duration	registration
When set to <i>DNSTTL</i> , the primary server is retried after a timeout equal to the DNSTTL configured for the server the handset is registered to (or via, for the outbound proxy scenario). When set to <i>registration</i> , the primary server is retried when the current working server's registration requires renewal. When set to <i>duration</i> , the primary server is retried after the amount of time defined by the <i>timeout</i> parameter (as shown in the next row). Note: When failback mode is set to <i>DNSTTL</i> or <i>duration</i> , re-registration with the primary server takes place only if the handset is idle (that is, the handset has no calls or active lines). If the timeout period expires and call activity is detected, failback will be retried every second.		
reg.x.server.y.failOver.failBack.timeout If you use proxy servers, set this parameter instead: reg.x.outboundProxy.failOver.failBack.timeout	0, 60 - 65535	3600
When failBack.mode is set to <i>duration</i> , the time in seconds after failing over to the current working server before the primary server becomes the first server to forward new requests to. If you set a value between 1 and 59, the timeout will be 60 seconds. If you set a value of 0, the primary server won't be selected as the first server to forward new requests to until a failover event occurs with the current working server.		

Table 10-20: All possible reg.x parameters used to set up server redundancy

Parameter	Permitted Values	Default
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.outboundProxy.failOver.failBack.mode	DNSTTL, registration, duration	registration
The mode for failover failback (overrides <code>reg.x.server.y.failOver.failBack.mode</code>). DNSTTL: the handset tries the primary server again after a timeout equal to the DNS TTL configured for the server that the handset is registered to.		

Parameter	Permitted Values	Default
<p>registration: the handset tries the primary server again when the registration renewal signaling begins.</p> <p>duration: the handset 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 handset waits this long after connecting to the current working server before selecting the primary server again. If 0, the handset 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 handset 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 handset 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 handset 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 handset 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 handset won't attempt to register with the secondary server, since the handset will assume that the primary and secondary servers share registration information.</p>		
reg.x.server.y.failOver.failBack.mode	DNSTTL, registration, duration	registration
<p>The mode for failover failback (this parameter overrides <code>voIpProt.server.x.failOver.failBack.mode</code>):</p> <p>DNSTTL – the handset tries the primary server again after a timeout equal to the DNS TTL configured for the server that the handset is registered to.</p> <p>registration – the handset tries the primary server again when the registration renewal signaling begins.</p> <p>duration – the handset tries the primary server again after the time specified by <code>reg.x.server.y.failOver.failBack.timeout</code>.</p>		
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 handset waits this long after connecting to the current working server before selecting the primary server again. If 0, the handset will not fail-back until a fail-over event occurs with the current server.</p>		
reg.x.server.y.failOver.failRegistrationOn	0 or 1	0
<p>When set to 1, and the <code>reRegisterOn</code> parameter is enabled, the handset 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 handset will attempt failback without first attempting to register with the primary server to determine if it has recovered.</p>		

Parameter	Permitted Values	Default
reg.x.server.y.failOver.onlySignalWithRegistered	0 or 1	1
When set to <i>1</i> , 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 handset 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 <i>0</i> , 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).		
reg.x.server.y.failOver.reRegisterOn	0 or 1	0
This parameter overrules the <code>voIpProt.server.x.failOver.reRegisterOn</code> . When set to <i>1</i> , the handset 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 <i>0</i> , the handset won't attempt to register with the secondary server, since the handset will assume that the primary and secondary servers share registration information.		
reg.x.tcpFastFailover	0 or 1	0
If <i>1</i> , failover occurs based on the values of <code>reg.x.server.y.retryMaxCount</code> or <code>voIpProt.server.x.retryTimeOut</code> .		

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). Spectralink handsets support standard IEEE 802.1X authentication. **Figure 10-1** shows the RADIUS server using the AP as the authenticator for the Spectralink 84-Series handsets.

Figure 10-1: A Typical 802.1X Network Configuration



Spectralink handsets support the following EAP authentication methods:

- EAP-PEAPv0/MSCHAPv2 (requires CA certificates)

- EAP-FAST (optional Protected Access Credential (PAC) file, if not using in-band provisioning)
- EAP-TLS (requires CA certificates)

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 10-21: Supporting 802.1X Authentication](#) to configure 802.1X Authentication.

For more information about certificates see [Downloading Certificates to a Spectralink Phone](#).



Web Info: EAP Authentication Protocol

For more information, see [RFC 3748](#), Extensible Authentication Protocol.

Summary

Parameter	Used to:
device.net.dot1x.identity	Specify the identity (username) for authentication
device.net.dot1x.password	Specify the password for authentication
device.net.dot1x.eapFastInBandProv	To enable EAP In-Band Provisioning for EAP-FAST
device.pacfile.data	Specify a PAC file for EAP-FAST (optional)
device.pacfile.password	Specify the optional password for the EAP-FAST PAC file

Table 10-21: Supporting 802.1X Authentication

Parameter	Permitted Values	Default
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	0
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.identity¹	string	Null
The identity (user name) for 802.1X authentication.		
Specify the 802.1X authentication method, where EAP-NONE 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.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>		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
device.pacfile.password¹	String	Null
EAP-FAST only, optional. The password for the PAC file.		

¹ Change causes handset to restart or reboot.

<tcplpApp/>

This parameter includes:

- <dhcp/>
- <dns/>
- <ice/>
- <keepalive/>

<dhcp/>

The DHCP parameters enable you to change how the handset reacts to DHCP changes.

Table 10-22: 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/> parameters provide 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 10-23: 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 handset directs DNS queries.		
tcplpApp.dns.altServer¹	Dotted-decimal IP address	Null
The secondary server to which the handset directs DNS queries.		
tcplpApp.dns.domain¹	String	Null
The handset's DNS domain.		

¹ Change causes handset to restart or reboot.

<ice/>

The <ice/> parameters enable you to set the STUN/TURN/ICE feature.

Table 10-24: 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 2013 or 2010, set to MSOCS to enable ICE.		
tcplpApp.ice.password	String	Null
Enter the password to authenticate to the TURN server.		
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	0 or 1	0
If 0, call admission control is disabled. If 1, call admission control is enabled		
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 TCP port number of the TURN server.		
tcplpApp.ice.username	String	Null
Enter the user name to authenticate to the TURN server.		

<keepalive/>

These parameters enable the configuration of TCP keep-alive on SIP TLS connections; the handset can detect a failure quickly (in minutes) and attempt to re-register with the SIP call server (or its redundant pair).

Table 10-25: TCP Keep-Alive Parameters

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
tcplpApp.keepalive.tcp.idleTransmitInterval	10 to 7200	30
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.		
tcplpApp.keepalive.tcp.noResponseTransmitInterval	5 to 120	20
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).		
tcplpApp.keepalive.tcp.sip.tls.enable	0 or 1	0
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.		

Tones <tones/>

This parameter describes configuration items for the tone resources available in the handset. It includes:

- <DTMF/>
- <chord/>

<chord/>

Chord-sets are the building blocks of sound effects that use 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 10-26: Chord Parameters

Parameter	Permitted Values
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.	

<i>Parameter</i>	<i>Permitted Values</i>
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.	

Web Configuration Utility

<httpd/>

The handset 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 10-27: 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 handset to restart or reboot.

Chapter 11: Special Use Cases

Certain situations require special handling and more involved provisioning. This chapter addresses these special use cases and the parameters that may be needed for a successful deployment.

Acoustic Echo Cancellation

Your Spectralink handset uses advanced acoustic echo cancellation (AEC) for handsfree operation using the speakerphone. The handset also supports headset echo cancellation. The handsets use both linear and non-linear techniques to aggressively reduce echo while permitting natural, full-duplex communication patterns.



Caution: Contact Spectralink Support Before Modifying Acoustic Echo Cancellation Parameters

Consult Spectralink 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 Spectralink handsets:

Table 11-1: Audio Codec Priority

<i>Supported Audio Codecs</i>	<i>Priority</i>
G.722	4
	5
G.711m-law	6
G.711a-law	7
G.729a, G.729b	8

The following table summarizes the audio codecs supported on Spectralink handsets:

Table 11-2: 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

<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 a-law	RFC 1890	64 Kbps	80 Kbps	8 Ksps	20 ms	3.5 KHz
G.711	RFC 1890	64 Kbps	80 Kbps	16 Ksps	20 ms	7 KHz
G.722.1	RFC 3047	16 Kbps 24 Kbps 32 Kbps	32 Kbps 40 Kbps 48 Kbps	16 Ksps	20 ms	7 KHz
G.729a	RFC 2833	8 Kbps	24 Kbps	8 Ksps	10 ms	8 KHz



Caution: New Zealand requirements

Re: The Optimum Packet Size for transmission through the Public Switched Telephone Network (PSTN)

Because of the extensive delay already experienced when calling cellular and international networks, Telecom Access Standards recommends the use of either 10 or 20mS packet length when passing packets through the PSTN. The use of G.711 codecs and 10 or 20mS packet length is critical to maintaining delay times which comply with PTC220 requirements for Customer Equipment (<50mS).



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.

Use the parameters in the following table to specify the priority for audio codecs on your Spectralink handsets.

Table 11-3: Audio Codec Priorities

<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.G722		4
voice.codecPref.G7221.16kbps		0
voice.codecPref.G7221.24kbps		0
voice.codecPref.G7221.32kbps		5
voice.codecPref.G729_AB		8

The priority of the codec. If 0 or Null, the codec is disabled. A value of 1 is the highest priority. If a handset does not support a codec, it will treat the setting as if it were 0 and not offer or accept calls with that codec.

Band Steering

Band steering is a feature that allows you to configure the handsets to use a preferred band during roaming.

If this preference is not configured, the 84-Series handset performs inter-band roaming between the 2.4GHz and 5GHz bands if both bands are enabled without any preference to one band over the other. The band of the access point with the strongest signal strength, as measured by the handset, is used.

Band steering is the use of a preferred band when selecting an AP. The handset uses the preferred band as long as telephone performance is not degraded by staying on the preferred band instead of using the non-preferred band.

Table 11-4: Band steering parameters

Parameter	Permitted Values	Default
preferredBandRoaming.band	noPreference prefer2_4GHz prefer5GHz	noPreference
<p>The configuration parameter, <code>preferredBandRoaming.band</code>, selects the preferred inter-band roaming mode. Allowable values are <code>noPreference</code> (the default value), <code>prefer2_4GHz</code>, or <code>prefer5GHz</code>. For inter-band roaming to function, both 2.4GHz and 5GHz bands must be enabled using existing <code>device.wifi.radio</code> parameters.</p>		
preferredBandRoaming.threshold	-65 to -40 db	-65
<p>The 84 Series handset makes band selections based on signal strength for three different conditions; strong signal strength, moderate signal strength, and low signal strength.</p> <p>If the measured signal strength on a channel in the preferred band is above <code>preferredBandRoaming.threshold + 10db</code>, the handset stays on the preferred band regardless of how much better the signal strength on the non-preferred band may be. There is no degradation in performance as long as the signal strength is strong so there is no reason to roam to the non-preferred band. The 84 Series handset may still roam between APs on the preferred band. The threshold is set by <code>preferredBandRoaming.threshold</code> which may be configured for RSSI signal strengths from -65db to -40db.</p> <p>While the default is -65db, Spectralink recommends that users experiment with values between -65db and -55db. The +10db is added to the threshold to prevent it from ping-ponging between two APs.</p> <p>On the other end of the signal strength range is the point where signal strength is so low the wireless performance is impacted. Below this threshold, the handset uses the access point with the best signal strength regardless of which band is available (i.e. uses the rules that existed prior to adding the band steering feature). Any user configured preference is ignored. This threshold is -75db and is not configurable.</p>		
preferredBandRoaming.bias	0 to 10 db	0
<p>The lower end of the signal strength range is the point where signal strength is so low the wireless performance is impacted. Below this threshold, the handset uses the access point with the best signal strength regardless of which band is available. Any user configured preference is ignored. This threshold is -75db and is not configurable.</p> <p>When the signal strength on the preferred band is moderate, between -75db and <code>preferredBandRoaming.threshold + 10db</code>, the 84 Series handset MAY use the preferred band. The handset uses the non-preferred band if the signal strength is significantly better than the preferred band. If the signal strength on the non-preferred band is only slightly better than on the preferred band, the 84 Series handset uses the preferred band.</p> <p>The handset software has 10db built into the roaming algorithm to prevent the telephone from constantly switching between two APs with similar signal strength. The handset will switch from a non-preferred band AP to a preferred band AP if the preferred band AP signal strength is greater than the signal strength of the non-preferred band AP. But to switch back to the non-preferred band AP, the non-preferred band AP must be at least 10db better than the preferred band AP. This amount can be increased up to 20db by using the bias parameter, thus strengthening the preferred band criteria. The <code>preferredBandRoaming.bias</code> is added to the preferred band signal strength before comparing it to the non-preferred band AP.</p>		

As an example, imagine the following scenario:

- The threshold is set to -60
- The bias is set to 5
- The band is set to prefer5GHz (a-band)

In this case, the following would be true:

- If there is a channel on the a-band that is at -50 or greater then the phone will always use the a-band and will never roam to the b-band. The value of -50 is the `threshold + 10` or $-60 + 10 = -50$.
- If the phone is on an AP on the a-band and the RSSI is less than -75, the phone will use the old rules to roam to a new AP. If it finds a new a band AP with sufficient signal strength it will use that AP, but may also roam to the b-band. At this point, it is just looking for the best AP it can find.
- If the phone is on the a-band and the candidate to roam to is also on the a-band, then normal roaming rules will apply.
- If the phone is on the a-band and the candidate to roam to is on the b-band, then the phone shall add the `bias` value into the hysteresis rules. The higher the `bias` the more it will try to stay on the preferred band (the a-band in this case) even if there are APs on the b-band that have better signal strength.

No other changes are made to the 84-Series handset roaming behavior. Setting `preferredBandRoaming.band` to `noPreference` results in the same roaming behavior as in previous versions of software. The 84-Series handset still does not actively roam in standby (to preserve battery life) but will use the inter-band roaming rules if the handset loses the network and must reacquire the network.

Bridged Line Appearance

Bridged line appearance connects calls and lines to multiple handsets. With bridged line appearance enabled, an active call displays simultaneously on multiple handsets in a group. By default, the answering handset has sole access to the incoming call—line seize. If the answering handset places the call on hold, that call becomes available to all handsets of that group. All call states—active, inactive, on hold—are displayed on all handsets of a group.



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 handset 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.

Summary

Parameter	Used to:
call.shared.disableDivert	Specify whether call diversion should be disabled by default on all shared lines
reg.x.type	Specify the per-registration line type (private or shared)
reg.x.thirdPartyName	Specify the shared line third-party name.
divert.x.sharedDisabled	Specify whether call diversion should be disabled on a specific shared line (overrides default)

Table 11-5: Enabling Bridged Line Appearance

Parameter	Permitted Values	Default
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.		
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.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.		
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.		

¹ Change causes handset to restart or reboot.

Local Digit Map

The handset has a local digit map feature that, when configured, will automatically call a dialed number, eliminating the need to press the **Dial** or **Send** softkey 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 handset's behavior. If a number ends with #, you can specify the handset's behavior, called trailing # behavior. You can also specify the digit map timeout, the period of time after you dial a number that the handset will wait before the call will be placed. The configuration syntax of the digit map is based on recommendations in section 2.1.5 of [RFC 3435](#).

Summary

Parameter	Used to:
dialplan.applyTo*	Apply a dial plan to dialing scenarios
dialplan.digitmap	Specify the digit map to use for the dial plan
dialplan.digitmap.timeOut	Specify the timeout for each segment of the digit map

Parameter	Used to:
<code>dialplan.impossibleMatchHandling</code>	Specify the behavior if an impossible dial plan match occurs
<code>dialplan.removeEndOfDial</code>	Specify if trailing # digits should be removed from digits sent out
<code>dialplan.routing.emergency.x.*</code>	Specify the details for emergency dial plan routing
<code>dialplan.routing.server.x.*</code>	Specify the server that will be used for routing calls
<code>dialplan.x.*</code>	Configure the same parameters as above for a specific registration (overrides the global parameters above)

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 handset 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. The following examples illustrate the semantics of the syntax:

- **R9R604Rxxxxxx**—Replaces 9 with 604
For example, a customer dials 91524810 – the digits sent to the PBX will be 6041524810
- **xxR601R600Rxx**—When applied to 1160122 gives 1160022
- **R9RRxxxxxx**—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.
- **RR604Rxxxxxx**—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.
- **xR60xR600Rxxxxxx**—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 any 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.

The following guidelines should be noted:

- The following letters are case sensitive: x, T, and R
- You must use only *, #, +, or 0-9 between the second and third R
- If a digit map does not comply with section 2.1.5 of RFC 3435, 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.
For example, R9Rxxxx would be invalid but RR9Rxxxx would be valid.
- If you use T in the left part of 'RRR' syntax, the digit map will not work. For example, R0TR322R will not work. The T can only be applied at the end of the digit string.

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.

All characters of a digit map must conform to the rules explained in section 2.1.5 of RFC3435. The following list defines the valid character types and characters usable within a digit map:

- Digit: A digit from "0" to "9".
- Timer: The symbol "T" matching a timer expiry.
- DTMF: A digit, a timer, or one of the symbols "A", "B", "C", "D", "#", or "*". Extensions may be defined.
- Wildcard: The symbol "x" which matches any digit ("0" to "9").
- Range: One or more DTMF symbols enclosed between square brackets "[" and "]").
- Subrange: Two digits separated by hyphen ("-") which matches any digit between and including the two. The subrange construct can only be used inside a range construct, i.e., between "[" and "]".
- Position: A period (".") which matches an arbitrary number, including zero, of occurrences of the preceding construct.
- Separator: An alternation operator or pipe ("|") is used to separate multiple digit map strings



Note: Using the alternation operator

When using the alternation operator—the vertical bar (|)—the elements must be enclosed in parenthesis. To specify A or B or C, the syntax should be (A|B|C).

- Miscellaneous: A comma (",") is used to return dial tone to the user during dialing

The Spectralink 84-Series handset utilizes a default digit map as part of the dial plan configuration. The following explains what each of the default digit mappings will match to:

- [2-9]11 – This string will match 211, 311, 411, 511, 611, 711, 811, or 911
- 0T – This string will match 0 and then wait for the inter-digit timeout to expire (3 seconds by default)
- +011xxx.T – This string will be used for international dialing within the United States and Canada. It will match 011 followed by any three digits (0-9) and then any additional number of digits until the inter-digit timeout expires (3 seconds by default)
- 0[2-9]xxxxxxxx – This string will be used for collect calling within the United States and Canada. It will match 0 plus a digit between 2 and 9 followed by nine additional digits (0-9)
- +1[2-9]xxxxxxxx – This string will be used for standard long distance dialing within the United States and Canada. It will match 1 plus a digit between 2 and 9 followed by eight additional digits (0-9)
- [2-9]xxxxxxxx – This string will be used for long distance dialing where ten digits are allowed. This will be dependent on the local PSTN provider as to whether it will be a valid dial string. Note that this is similar to ten digit dialing with cellular telephones. It will match a digit between 2 and 9 followed by nine additional digits (0-9)
- [2-9]xxxT – This string will be used for local dialing in situations where area codes and local exchange codes are not required. This string may also match internal extension based dialing depending on the digits being dialed. It will match a digit between 2 and 9 followed by three additional digits (0-9) until the inter-digit timeout expires (3 seconds by default)

Please note that the default dial plan may not be sufficient for your particular needs. For example, many customers require a digit be dialed to access an outside line, such as 9. In this case they may not be able to dial a sufficient number of digits before the call attempts to complete.

- For example, a customer is attempting to dial 913035551155. With the default digit map this would match to [2-9]xxxxxxxx. This would mean that after the customer had dialed 9130355511 the call would attempt to complete. However, this would be an incomplete dial string so it would fail to connect.

To remedy the issue it is recommended that customers consider their dial plan carefully. It may be prudent to alter the default digit map as well to a more simplified format. One such example would be to set the digit map to x.T and remove all other strings. This string would match all possible dialed digits as it will allow for one digit (0-9) followed by any number of additional digits (0-9). The T at the end will mean that the user will have 3 seconds, by default, between each digit before the handset stops collecting digits and attempts to complete the call.

Many customer environments rely on complex dial plans within the PBX to handle call routing and class of service requirements to prevent undesired dialing by users. Because of this it may

not be necessary to further restrict the dial plan at the handset level. If this is the case then a simplified digit map, as described, would be ideal.

Table 10-6: 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 softkey to place a call. If 1, the dial plan applies to calls placed using the Dial softkey.		
dialplan.applyToUserSend¹	0 or 1	1
If 0, the dial plan does not apply to calls placed when the user presses the Send softkey to place a call. If 1, the dial plan applies to calls placed using the Send softkey.		
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
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 handset to automatically initiate calls to numbers that match a digit map pattern.		
dialplan.digitmap.timeOut¹	string of positive integers separated by ' '	3 3 3 3 3 3
Specify a timeout in seconds for each segment of digit map. After you press a key, the handset 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.		
dialplan.filterNonDigitUriUsers¹	0 or 1	0
If 0, do not filter out (+) in the dial plan. If 1, filter out (+) from the dial plan.		
dialplan.impossibleMatchHandling¹	0, 1 or 2	0
This parameter applies to digits entered in dial mode. Users are in dial mode after going off-hook on the handset or headset, or after pressing the New Call key. Users are not in dial mode when on-hook 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, the handset gives the reorder tone when the impossible match occurs.		
If set to 2, allow user to accumulate digits and dispatch call manually with the Send softkey.		

Parameter	Permitted Values	Default
dialplan.removeEndOfDial¹	0 or 1	1
If set to 1, strip trailing # digit from digits sent out.		
dialplan.routing.emergency.outboundIdentity	SIP, secure SIP, or TEL URI	Null
The identity used to identify your handset when you place an emergency call from your handset. A valid SIP, secure SIP, or TEL URI. The string may be 10 to 25 characters in length.		
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
<p>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.</p> <p>description: The label or description for the emergency address</p> <p>server.y: The index representing the server to use for emergency routing (dialplan.routing.server.x.address where x is the index).</p> <p>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.</p> <p>Note: Blind transfer for 911 (or other emergency calls) may not work if registration and emergency servers are different entities.</p>		
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 handset to restart or reboot.

Per-registration dial plan configuration is also supported. The descriptions for each parameter are in the table below. The parameters listed in this table overrule the parameters in the previous table for registration x, where x is the registration number (for example, dialplan.x.applyToTelUriDial overrules dialplan.applyToTelUriDial for registration x):

Table 10-7: Per-Registration Dial Plan (Digit Map) Parameters

Parameter	Permitted Values	Default
dialplan.x.applyToCallListDial ¹	0 or 1	1
dialplan.x.applyToDirectoryDial ¹	0 or 1	0
dialplan.x.applyToForward	0 or 1	0

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
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, UDPOOnly, TLS, TCPOnly	DNSnaptr

¹ Change causes handset to restart or reboot.

Location Values for E.911 Services

The values you enter for these Lync Server-only parameters will be used by E.911 services.

Table 11-8

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
locInfo.x.label Enter a label for your location.	String	Null
locInfo.x.country Enter the country the handset is located in.	String	Null
locInfo.x.A1 Enter the national subdivision the handset is located in, for example, a state or province.	String	Null

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
locInfo.x.A3 Enter the city the handset is located in.	String	Null
locInfo.x.PRD Enter the leading direction of the street location.	String	Null
locInfo.x.RD The name of the road or street the handset is located on.	String	Null
locInfo.x.STS Enter the suffix of the name used in locInfo.x.RD, for example, Street, Avenue.	String	Null
locInfo.x.POD Enter the trailing street direction, for example SW.	String	Null
locInfo.x.HNO Enter the street address number of the handset's location.	String	Null
locInfo.x.HNS Enter a suffix for the street address used in locInfo.x.HNS, for example, ^A or ½.	String	Null
locInfo.x.LOC Enter any additional information that identifies the location.	String	Null
locInfo.x.NAM Enter a name for the location, for example, a business name, an occupant, a resident.	String	Null
locInfo.x.PC Enter the postal code of the location.	String	Null

Real-Time Transport Protocol Ports

You can configure the handset 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 handset receives from a non-negotiated IP address or a non-negotiated port.

You can configure the handset 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 handset'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 handset's RTP port range. Since the handset supports conferencing and multiple RTP streams, the handset can use several ports concurrently. Consistent with RFC 1889, the next-highest odd-numbered port is used to send and receive RTP.

The handset 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 handset 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.

Summary

Parameter	Used to:
<code>tcpIpApp.port.rtp.filterByIp</code>	Filter RTP packets by IP address
<code>tcpIpApp.port.rtp.filterByPort</code>	Filter RTP packets by port
<code>tcpIpApp.port.rtp.forceSend</code>	Force-send packets on a specified port
<code>tcpIpApp.port.rtp.mediaPortRangeStart</code>	Set the starting port for RTP packet port range

Table 11-9: Configuring Real-Time Transport Protocol Ports

Parameter	Permitted Values	Default
<code>tcpIpApp.port.rtp.filterByIp</code>¹	0 or 1	1
IP addresses can be negotiated through the SDP protocols. If set to 1, the handset rejects RTP packets that arrive from non-negotiated IP addresses.		
<code>tcpIpApp.port.rtp.filterByPort</code>¹	0 or 1	0
Ports can be negotiated through the SDP protocol. If set to 1, the handset will reject RTP packets arriving from (sent from) a non-negotiated port.		
<code>tcpIpApp.port.rtp.forceSend</code>¹	0 to 65535	0
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.		
<code>tcpIpApp.port.rtp.mediaPortRangeStart</code>¹	even integer 1024 to 65440	2222
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). <i>Note:</i> Ensure that there is no contention for port numbers. For example, do not use 5060 (default port for SIP).		

¹ Change causes handset to restart or reboot.

Shared Line Appearances

With the shared call appearance feature enabled, an active call displays simultaneously on multiple handsets in a group. By default, the answering handset has sole access to the incoming call, called line seize. You can enable another handset in the group the ability to enter a conversation, called a barge in. If the answering handset places the call on hold, that call becomes available to all handsets of that group. All call states of a call —active, inactive, on hold—are displayed on all handsets of a group.

This feature is dependent on support from a SIP call server. To enable shared call appearances on your handset, you will need to obtain a shared line address from your SIP service provider.

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 handset 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 handset to ask to seize the line

Summary

Parameter	Used to:
reg.x.address	specify the shared line address.
reg.x.type	specify the line type as shared.
reg.x.*	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.
call.shared.*	disable call diversion, expose auto-holds, resume with one touch, or play a tone if line-seize fails.
volpProt.SIP.specialEvent.lineSeize.no nStandard	specify standard or non-standard behavior for processing a line-seize subscription for mutual exclusion.
divert.x.sharedDisabled	specify per-registration whether diversion should be disabled on shared lines.

Table 11-10: Shared Call Appearances

Parameter	Permitted Values	Default
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.seizeFailReorder¹	0 or 1	1
If set to 1, play re-order tone locally on shared line seize failure.		
volpProt.SIP.specialEvent.lineSeize.nonStandard¹	0 or 1	1
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.		
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.		

¹ Change causes handset to restart or reboot.

Static DNS Cache

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 is to statically configure a set of DNS NAPTR SRV and/or A records into the handset.

When a handset is configured with a DNS server, it will behave as follows by default:

- The handset 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 handset 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 handset is not configured with a DNS server, it will behave as follows:

- When the handset 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](#).

Summary

<i>Parameter</i>	<i>Used to:</i>
reg.x.server.y.*	Specify the call server used for this registration
dns.cache.A.x.*	Specify the DNS A address, hostname, and cache time interval (ttl)
dns.cache.NAPTR.x.*	Specify the DNS NAPTR parameters, including: name, order, preference, regexp, replacement, service, and ttl
dns.cache.SRV.x.*	Specify DNS SRV parameters, including: name, port, priority, target, ttl, and weight

Table 11-11: Configuring the Static DNS Cache

<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 overrule the parameters specified in <code>voIpProt.server.*</code>. <i>Notes:</i> If this parameter is set, it overrules 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>		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
dns.cache.A.x.address IP address.	dotted-decimal IP version 4 address	Null
dns.cache.A.x.name Hostname	valid hostname	Null
dns.cache.A.x.ttl The TTL describes the time period the handset 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.	300 to 536870912 (2^29), seconds	300
dns.cache.NAPTR.x.flags The flags to control aspects of the rewriting and interpretation of the fields in the record. Characters are case-sensitive. At this time, only 'S', 'A', 'U', and 'P' are defined as flags. See RFC 2915 for details of the permitted flags.	A single character from [A-Z, 0-9]	Null
dns.cache.NAPTR.x.name The domain name to which this resource record refers.	domain name string	Null
dns.cache.NAPTR.x.order An integer specifying the order in which the NAPTR records must be processed to ensure the correct ordering of rules.	0 to 65535	0
dns.cache.NAPTR.x.preference 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.	0 to 65535	0
dns.cache.NAPTR.x.regexp This parameter is currently unused. Applied to the original string held by the client. The substitution expression is applied in order to construct the next domain name that will be looked up. The grammar of the substitution expression is given in RFC 2915 .	string containing a substitution expression	Null
dns.cache.NAPTR.x.replacement The next name to query for NAPTR records depending on the value of the flags field. It must be a fully qualified domain-name.	domain name string with SRV prefix	Null
dns.cache.NAPTR.x.service Specifies the service(s) available down this rewrite path. For more information, see RFC 2915 .	string	Null
dns.cache.NAPTR.x.ttl The TTL describes the time period the handset 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.	300 to 536870912 (2^29), seconds	300
dns.cache.SRV.x.name The domain name string with SRV prefix.	domain name string with SRV prefix	Null
dns.cache.SRV.x.port The port on this target host of this service. For more information, see RFC 2782 .	0 to 65535	0
dns.cache.SRV.x.priority The priority of this target host. For more information, see RFC 2782 .	0 to 65535	0

Parameter	Permitted Values	Default
dns.cache.SRV.x.target	domain name string	Null
The domain name of the target host. For more information, see RFC 2782 .		
dns.cache.SRV.x.ttl	300 to 536870912 (2^29), seconds	300
The TTL describes the time period the handset 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.cache.SRV.x.weight	0 to 65535	0
A server selection mechanism. For more information, see RFC 2782 .		

Example Static DNS Cache Configuration

The following examples show you how to configure the static DNS cache.

Example 1

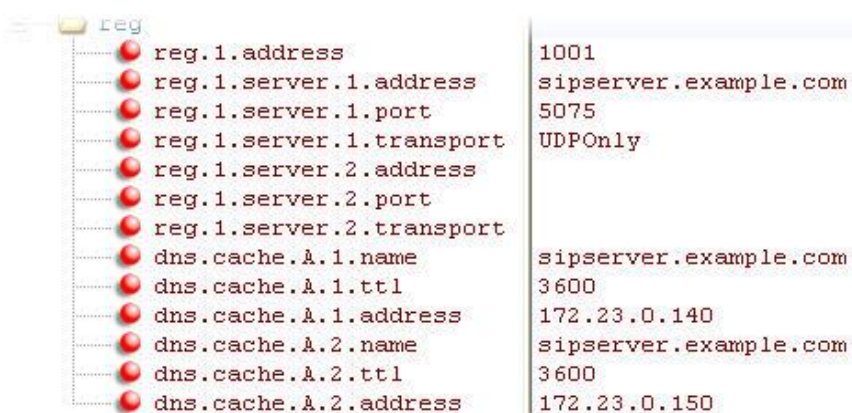
This example shows how to configure static DNS cache using A records IP addresses in SIP server address fields.

When the static DNS cache is not used, the **site.cfg** configuration will look as follows:



reg.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.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 84-Series 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.y.address` but not SRV. In this case, the static DNS cache on the handset 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:

 reg	
reg.1.address	1002@sipserver.example.com
reg.1.server.1.address	primary.sipserver.example.com
reg.1.server.1.port	5075
reg.1.server.1.transport	UDPOOnly
reg.1.server.2.address	secondary.sipserver.example.com
reg.1.server.2.port	5075
reg.1.server.2.transport	UDPOOnly

When the static DNS cache is used, the **site.cfg** configuration will look as follows:

 reg	
reg.1.address	1002
reg.1.server.1.address	sipserver.example.com
reg.1.server.1.port	
reg.1.server.1.transport	UDPOOnly
reg.1.server.2.address	
reg.1.server.2.port	
reg.1.server.2.transport	
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



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:



The image displays two identical screenshots of a configuration interface. Each screenshot shows a tree view with a folder icon and the label 'reg'. To the right of the tree is a table of configuration parameters. The parameters are listed in a column on the left, and their corresponding values are listed in a column on the right. The values are: '1002@sipserver.example.com' for 'reg.1.address', '172.23.0.140' for 'reg.1.server.1.address', '5075' for 'reg.1.server.1.port', 'UDPOnly' for 'reg.1.server.1.transport', '172.23.0.150' for 'reg.1.server.2.address', '5075' for 'reg.1.server.2.port', and 'UDPOnly' for 'reg.1.server.2.transport'.

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:

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.regex	
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.

Using Static DNS Cache for Redundancy

Failover redundancy can only be utilized when the configured IP server hostname resolves (through SRV or A record) to multiple IP addresses. Unfortunately, some customer's are unable to configure the DNS to take advantage of failover redundancy.

The solution is to provide the ability to statically configure a set of DNS NAPTR SRV and/or A records into the handset.

When a handset is configured with a DNS server, it will behave as follows by default:

- An initial attempt to resolve a hostname that is within the static DNS cache, for example to register with its SIP registrar, results in a query to the DNS.

- 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 handset is not configured with a DNS server, it will behave as follows

- An attempt to resolve a hostname that is within the static DNS cache 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, go to <http://tools.ietf.org/html/rfc2308>.

Configuration File Changes

Configuration changes can be performed centrally at the boot server:

Central (boot server)	Configuration file: sip.cfg	Specify DNS NAPTR, SRV, and A records for use when the handset is not configured to use a DNS server. For more information, refer to DNS Cache <dns/>.
--------------------------	--------------------------------	---

DNS Cache <dns/>

In the tables below, a maximum of 12 entries of NAPTR, SRV, and A record can be added.

This attribute includes:

- • NAPTR <NAPTR/> attribute
- • SRV <SRV/>
- • A <A/>

NAPTR <NAPTR/>

Attribute	Permitted Values	Default	Interpretation
dns.cache.NAPTR.x.name	domain name string	Null	The domain name to which this resource record refers.
dns.cache.NAPTR.x.ttl	0 to 65535, seconds	300	Specifies the time interval that the resource record may be cached before the source of the information should again be consulted.
dns.cache.NAPTR.x.order	0 to 65535	0	A 16-bit unsigned 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 being processed before high numbers.

Attribute	Permitted Values	Default	Interpretation
dns.cache.NAPTR.x.flags	string	Null	Flags to control aspects of the rewriting and interpretation of the fields in the record. Flags are single characters from the set [A-Z, 0-9]. The alphabetic characters are case insensitive. At this time only four flags, "S", "A", "U", and "P" are defined. For more information, go to http://tools.ietf.org/html/rfc2915 .
dns.cache.NAPTR.x.service	string	Null	Specifies the service(s) available down this rewrite path. For more information, go to http://tools.ietf.org/html/rfc2915 .
dns.cache.NAPTR.x.regex	string	Null	A string containing a substitution expression that is applied to the original string held by the client in order to construct the next domain name to lookup. The grammar of the substitution expression is given in RFC 2915. Note: This attribute is currently not used.
dns.cache.NAPTR.x.replacement	domain name string with SRV prefix	Null	The next name to query for NAPTR, SRV, or address records depending on the value of the flags field. It must be a fully qualified domain-name.

SRV <SRV/>

This configuration attribute is defined as follows:

Attribute	Permitted Values	Default	Interpretation
dns.cache.SRV.x.name	domain name string	Null	The domain name string with SRV prefix.
dns.cache.SRV.x.ttl	0 to 65535, seconds	300	Specifies the time interval that the resource record may be cached before the source of the information should again be consulted.
dns.cache.SRV.x.priority	0 to 65535	0	The priority of this target host. For more information, go to http://tools.ietf.org/html/rfc2782 .
dns.cache.SRV.x.weight	0 to 65535	0	A server selection mechanism. For more information, go to http://tools.ietf.org/html/rfc2782 .
dns.cache.SRV.x.port	0 to 65535	0	The port on this target host of this service. For more information, go to http://tools.ietf.org/html/rfc2782 .
dns.cache.SRV.x.target	domain name string	Null	The domain name of the target host. For more information, go to http://tools.ietf.org/html/rfc2782 .

A <A/>

This configuration attribute is defined as follows:

Attribute	Permitted Values	Default	Interpretation
dns.cache.A.x.name	valid hostname	Null	Hostname

Attribute	Permitted Values	Default	Interpretation
dns.cache.A.x.ttl	0 to 65535	300	Specifies the time interval that the resource record may be cached before the source of the information should again be consulted.
dns.cache.A.x.address	dotted-decimal IP version 4 address	Null	IP address that hostname <code>dns.cache.A.x.name</code> maps to.

Examples

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 configuration would look as follows:

```
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 configuration would look as follows:

```
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"
```



Parameter order

Above addresses are presented to SIP application 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 `server.X.address` but not SRV. In this case, the static DNS cache on the handset provides SRV records. For more information, go to <http://tools.ietf.org/html/rfc3263> .

When the static DNS cache is not used, the configuration would look as follows:

```
reg.1.address="1002@sipserver.example.com"
reg.1.server.1.address="primary.sipserver.example.com"
reg.1.server.1.port="5075"
reg.1.server.1.transport="UDPOnly"
reg.1.server.2.address="secondary.sipserver.example.com"
reg.1.server.2.port="5075"
reg.1.server.2.transport="UDPOnly"
```

When the static DNS cache is used, the configuration would look as follows:

```
reg.1.address="1002"
reg.1.server.1.address="sipserver.example.com"
reg.1.server.1.port=""
reg.1.server.1.transport="UDPOnly"
reg.1.server.2.address=""
reg.1.server.2.port=""
reg.1.server.2.transport=""
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"
```



Reason for Null setting

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 `server.X.address` .

When the static DNS cache is not used, the configuration would look as follows:

```

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 configuration would look as follows:

```

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"

```



Reason for Null setting

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.

Voice Activity Detection

The purpose of voice activity detection (VAD) is to detect periods of silence in the transmit data path so the handset doesn't have to transmit unnecessary data packets for outgoing audio. This process conserves network bandwidth. VAD must be turned on in the PBX for it to work properly.

For compression algorithms without an inherent VAD function, such as G.711, the handset 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 handset 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.

These settings control the performance of the voice activity detection (silence suppression) feature.

Table 11-12: Voice Activity Detection (VAD) Parameters

Parameter	Permitted Values	Default
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 voice.vadEnable.		
<ul style="list-style-type: none"> If voice.vadEnable 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 voice.vadEnable 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 handset to restart or reboot.

Part IV: Troubleshooting and Maintaining your Deployment

Part IV provides you with the information you need to troubleshoot issues with your Spectralink 84-Series handsets and for basic, advanced, audio, user and telephone security features.

Part IV consists of the following chapters:

- [Chapter 11: Troubleshooting Your Spectralink handsets](#)
- [Chapter 12: Miscellaneous Maintenance Tasks](#)

Chapter 12: Troubleshooting Your Spectralink Handsets

This chapter shows you some tools and techniques for troubleshooting Spectralink handsets running Spectralink Software. The handset 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
- Network Protocols and Ports Used by Spectralink handsets

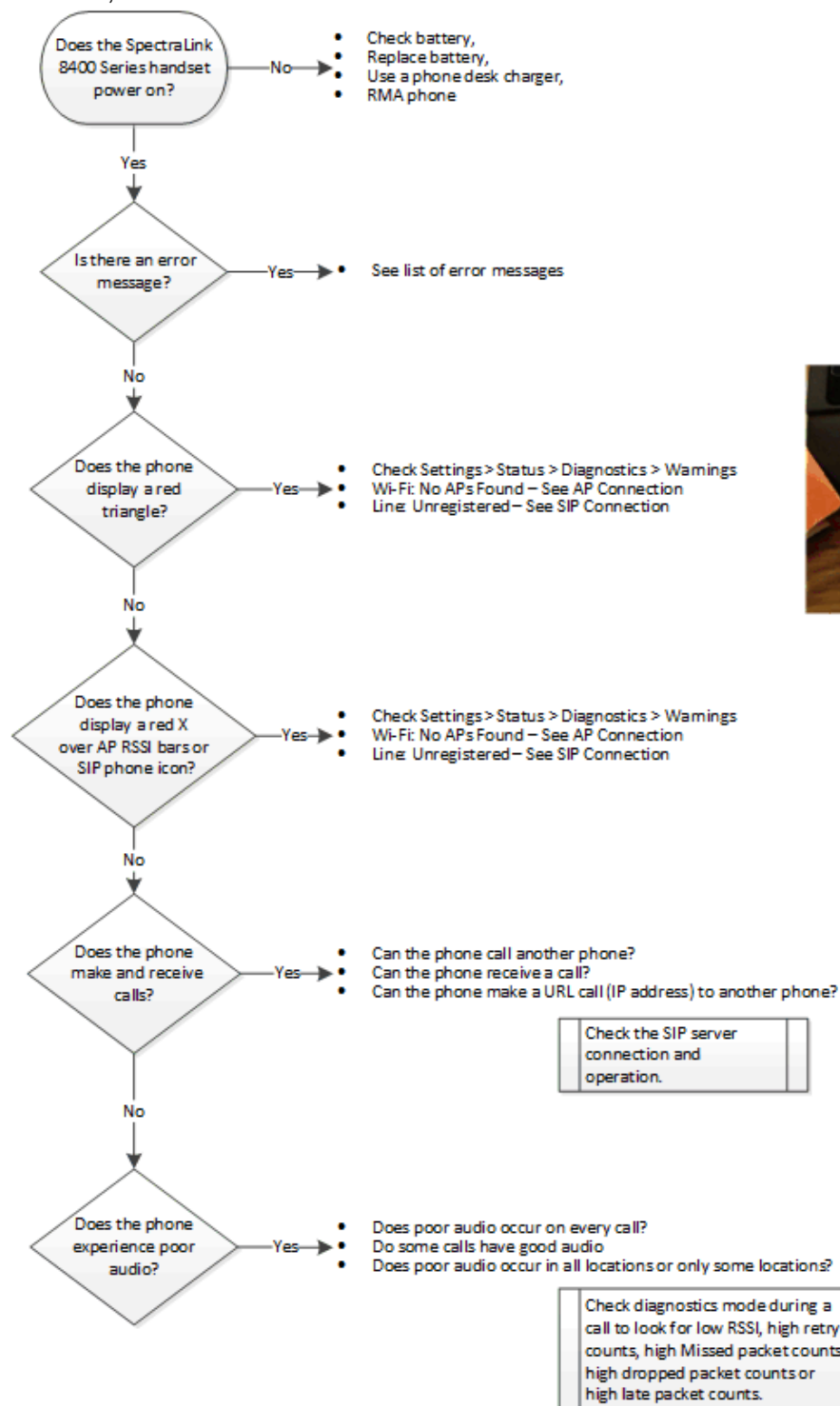
This chapter also addresses handset issues, likely causes, and corrective actions. Issues are grouped as follows:

- Power and Startup Issues
- Dial Pad Issues
- Screen and System Access Issues
- Calling Issues
- Display Issues
- Audio Issues
- Licensed Feature Issues
- Upgrading Issues

Review the latest *Spectralink Software Release Notes* on the support site for known problems and possible workarounds. If a problem is not listed in this chapter or in the latest *Release Notes*, contact your Certified Spectralink Reseller for support.

Troubleshooting Flow Diagram

A troubleshooting flow diagram provides an easy to follow list of symptoms to facilitate problem isolation, data collection and resolution.



Understanding Error Message Types

Several types of errors can occur while the handset is booting. If an error occurs, the handset will inform you by displaying an error message. Errors can affect how the handset boots up. If the error is fatal, the handset will not be able to boot until the error is resolved. If the error is recoverable, the handset will continue to boot but the handset's configuration may change or be incomplete.

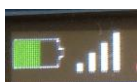
Updater Error Messages

Most of the following errors will be logged to the handset's boot log. However, if you are having trouble connecting to the provisioning server, the handset will likely not be able to upload the boot log <MAC-boot.log> file to its provisioning server for later analysis.

Failed to get boot parameters via DHCP

The handset does not have an IP address and therefore cannot boot. Check the following items:

- Is the 84-Series handset connected to the WLAN?
 - Does the AP signal strength indicator, at the top left of the handset display, show connected to the WLAN, as shown below, or does it have a red X across the AP signal strength bars?



- A red X across the AP signal strength bars indicates the handset is not connected to the WLAN. In this case, ensure the handset has the proper parameters set. These are configured either during the initial SLIC configuration with the USB cable or thru the handset keypad interface. Ensure the proper radio is turned on and WLAN security with the AP is turned on and available.
- Check the DHCP configuration.
 - **Settings> Status> Network> TCP/IP Parameters.** Is DHCP enabled or disabled.
 - Does the handset have an IP address on the right network subnet?
 - Check the IP, Subnet Mask & IP Gateway. Are they as expected?
 - If not, check with the DHCP server administrator to ensure that DHCP is providing the 84-Series handset with the proper settings.
 - If yes, are the neighbor addresses in the right subnet as expected? **Settings> Status> Network> Neighbors**
 - Can the handset ping its provisioning and SIP server IP addresses? **Settings> Status> Diagnostics> Network> Ping> Enter IP address to ping> Start**

Application <file name> is not compatible with this handset!

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 handset. 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.

Do other 84-Series handsets work with this version of software?

Updater has an incorrect signature

An error is detected by a .ld file just loaded to the handset. While customers load and run General Availability (GA) code from the Spectralink.com web site, there are a few special 84-Series handsets which use special engineering code. This error means the wrong code (GA code on engineering handset or vice versa) is being loaded on the handset. Ensure the correct *.sip.ld code is loaded to the handset. Download the latest code from the www.spectralink.com web site to get GA code.

Could not contact boot server using existing configuration

The handset could not contact the provisioning server, but the causes may be numerous. It may be related to DHCP configuration, or it could be a problem with the provisioning server itself. The handset can recover from this error so long as it previously downloaded a valid application BootROM image and all of the necessary configuration files.

- Check the provisioning boot server IP address known by the handset at **Settings> Status> Platform> Configuration**. This screen shows Boot server IP and protocol.
- Determine if the boot server IP address can be Pinged from the handset by going to **Settings> Status> Diagnostics> Network> Ping**. Enter the IP address and press start.
- Check the Boot server in the load protocol log file to see if there are any errors or reasons why the configuration files do not load.
- Ensure there are no network fire walls blocking the 84-Series handset from accessing the boot server.

Error, application is not present!

This message indicates that the handset has no application (operating handset code) stored in firmware or memory, that the handset could not download an application, and that the handset cannot boot. To resolve this issue, you must download compatible Spectralink Software to the handset using one of the supported provisioning protocols. You need to resolve the issue of connecting the handset to the provisioning server and provide a compatible software image on the provisioning server. This error is fatal, but recoverable. Check the following:

- Check the boot server IP address at **Settings> Status> Platform> Configuration**. Ensure the boot server is operational and available.

- Check the protocol used by the handset at the above menu. Ensure the FTP/TFTP server is running on the boot server.
- Check the boot server log for the proper protocol to determine if the handset contacted the server and asked for a sip.ld.
- Check the parameter APP_FILE_PATH="sip.ld" in the 000000000000.cfg or <mac>.cfg file to ensure a load file is specified.
- Is the proper sip.ld file available at the FTP/TFTP server location/directory specified?

Spectralink Software Error Messages

The warning notification feature 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 error icon in the status bar on the idle display, as shown next:



- A persistent list of current warnings, which can be viewed from **Settings> Status> Diagnostics> Warnings**

Wi-Fi: No APs Found

This message displays on Spectralink handsets if the handset is unable to connect to an access point (AP) on the wireless network. Check the following:

- Ensure the initial configuration parameters are set properly in the handset using the USB cable or keypad to set domain, SSID, turn on the radio, set WLAN security, set DHCP or provide a static IP and specify the boot server IP address and protocol.
- Ensure an AP with the proper SSID is available
 - Check at **Settings> Status> Diagnostics> WiFi Stats**
 - » Screen 2/6 General Info shows the SSID configured in the handset
 - » Screen 3/6 AP List shows available APs
 - Check Site Survey mode at: **Settings> Advanced Settings> [enter password]> Administration Settings> Diagnostics> Run Site Survey**
 - » The Current Configuration screen shows SSID, Regulatory Mode and frequency bands enabled
 - » Press Start to see available APs, Channel, RSSI and SSID for the configuration in the handset
 - » Press All to see all available APs for all (any) SSIDs

Network link is down

Since the Spectralink handsets do not have an LED indicating network LINK status like many networking devices, link failures are indicated with a message.

The 'Network link is down' message will be shown on the screen whenever the handset is not in the menu system and will persist until the link problem is resolved. Call related functions and the softkeys and line keys are disabled when the network is down; however the menu system works.

Check the top left corner of the 84-Series display to determine if the AP signal strength bars have a red X across them.

If there is a red X, the handset is not connected to an AP.

Has the handset connected to the WLAN before or is this the first failure of this kind? Is the AP operational?

Ensure the handset is configured properly for the WLAN (SSID, security, RF radio and channels,

If there is no red X then the handset has connected to an AP.

This means the handset and AP agree for SSID, wireless security, RF radio and channels, etc.

Does the handset have an IP address?

If DHCP is enabled then it must supply an IP address to the handset.

If DHCP is disabled, then the handset will have a static IP address assigned.

Check handset for an IP address at **Settings> Status> Network> TCPIP Parameters**

Can it ping the boot server, SIP server or another 84-Series handset?

The Ping function is at: **Settings> Status> Diagnostics> Network> Ping**. Enter IP address and press Start.

Did the handset download configuration files when it booted?

Check **Settings> Status> Platform> Configuration** scroll down to see file names and parameters loaded from each one

Config file errors

Config file error: Files contain invalid params: <filename1>, <filename2>,...

Config file error: <filename> contains invalid params.

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 the *Deployment Guide* for details.

Navigate to **Settings> Status> Platform> Configuration** then scroll down to see file names of the files that loaded and number of parameters loaded from each file (Errors, Duplicates or OK).

Are common files loaded properly by other handsets?

Did the problem just start? What has changed? Have the files been revised recently?

Insufficient Bandwidth

This message displays if a Spectralink handset has a poor network connection or the AP does not have enough bandwidth available to handle this handset.

This error could go with the Admission Control setting. Generally, the handset will require enough bandwidth to handle (standby) control frames and in call signaling and audio frames. When the AP has little bandwidth available because of other clients, with Admission Control enabled, it will inform the handset there is not enough bandwidth available. The handset will roam to find another AP, if possible. If there is no other APs available the handset is not able to recover.

Move the handset to another area with required bandwidth or add RF signaling with new APs so there is enough bandwidth available in this and other areas of the facility.

- Does the handset work in other areas of the facility (other APs)?
- Does the failure occur in one area or the same area each time?

Invalid Regulatory Domain

This message will display on Spectralink 84-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* softkey 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.

- Set the Regulatory Domain to the proper domain number and try again.
- Check Site Survey mode at: **Settings> Advanced Settings> [enter password]> Administration Settings> Diagnostics> Run Site Survey**
 - The Current Configuration screen shows SSID, Regulatory Mode and frequency bands enabled
- Are there WLAN APs available with this SSID, security, frequency bands and channels?

Invalid Regulatory Domain Setting

This message will display on Spectralink 84-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 restriction limits, 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.

Line: Unregistered

This message displays if a line fails to register with the call server. Check the following:

- Does the handset have the proper SIP server IP address? Navigate to **Settings> Status> Lines> Server-1: IP Address**
- Do the parameters in [the config file with extension parameters] match the SIP server for this extension?
- Are there errors in [the config file with extension parameters]? Navigate to **Settings> Status> Platform> Configuration**
 - Config: files loaded to the handset
 - Web: over-ride parameters loaded
 - Local: for over-ride parameters loaded
 - [Extension parameters]: Are there errors or duplicates? Is this extension set up and enabled in the SIP server?

Login credentials have failed. Please update them if information is incorrect.

This message displays when the user enters incorrect login credentials (**Settings> Basic Settings> Login Credentials**).

- Ensure the username entered matches a .cfg file which has extension parameters.
Example: User enters Username: Sallyj. A Sallyj.cfg file must exist. User enters a password. The Sallyj.cfg file must have a password parameter which matches the password entered by the user. Finally, the Sallyj.cfg file must have parameters which allow the connection to the SIP server.
- Check the files loaded at: **Settings> Status> Platform> Configuration**
 - Config: files loaded to the handset
 - Web: over-ride parameters loaded which cause a failure
 - Local: over-ride parameters loaded which cause a failure
 - [Extension parameters]: for proper settings to allow the connection. Are there errors or duplicates?
- Check the boot server and protocol log file to determine which file is asked for, if it exists and if it is loaded to the handset.

Time/Date out of sync

This message indicates the SNTP server and services are not available or could not be contacted. It could also be the configuration files do not specify the SNTP server.

Ensure the SNTP server is available and config files are setup properly.

Missing files, config reverted

This message displays when errors in the configuration or a failure to download the configuration files force the handset to revert to its previous (known) condition with a complete set of configuration files. This will also display if the files listed in the 000000000000.cfg or **<MAC Address>.cfg** file are not present on the provisioning server. Check the following:

- Navigate to **Settings> Status> Platform> Configuration** scroll down to see files list.
- Check the CONFIG_FILES= parameter in the 000000000000.cfg or <mac>.cfg file to see if it has the proper files specified, the files are available and the boot server is available.
- Check the boot server and protocol log file to determine which file is asked for and if it is loaded to the handset.

Network Authentication Failure

This message displays if 802.1X authentication between the Spectralink handset and WLAN AP fails. The codes shown in the following table will display on the handset's screen if the **Details** softkey is pressed. They can also be found 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.
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 handset, "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 handset, 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.

Status Menu

Debugging of a single 84-Series handset may be possible by examining the handset's status menu. Press the right arrow key, select **Settings> Status**, and press the OK key to view the Status menu. Scroll to one of the Status menu items and press the OK key. Each of the menu items are explained next.



Troubleshooting: I can't find the Status menu on my Spectralink handset

To view the **Status** menu on a Spectralink handset use the left or right arrow key to navigate to **Settings> Status> OK**.

Under the **Platform** menu, you can get details on the handset's serial number, MAC address, the current IP address, the Updater version, the application version, the name of the SIP.Id file loaded, the names of the config files loaded, and the IP address and protocol of the provisioning server along with Errors, Duplicates and OK parameters per file.

In the **Network** menu, you can find information about the TCP/IP Setting, DHCP, handset IP, subnet mask, default gateway, SNTP, DNS info, , and up time and statistics on packets sent and received since last boot. The Neighbors screen shows the MAC and IP address of neighbor nodes in this subnet.

The **Lines** menu will show you details about the status of each line that has been configured on the handset as well as Server IP address.

The **Diagnostics** menu offers a series sub menus to test the following:

- Hardware tests to verify correct operation of the microphone, speaker, handset, keypad, display, LED and third party headset, if present. You can also test that each of the keys on the handset is working, and display the function assigned to each of the keys in the configuration.
- Graphs show CPU load, Network load and Memory Usage
- Media Statistics shows Audio codec packet stats and Jitter
- Network allows Ping tests and Trace Route tests
- WiFi Stats shows six screens displaying [1/6] packet counts, [2/6]General Info (SSID, Last data Rate, Tx Power), [3/6]AP List (up to four APs with MAC, channel and RSSI), [4/6]Re/Association Stats, [5/6]WLAN Security stats, [6/6]EAP Info including 802.11n if disabled.
- Warnings menu shows warning messages related to handset operation, connection to the WLAN, SIP server, etc.
- The Licenses menu shows installed license information
- The Location menu shows location information.

Log Files

Spectralink handsets 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 handset's home directory or a user-configurable directory. You can also configure a handset to send log messages to a syslog server.

There is one log file for the Updater and one for the Spectralink Software. When a handset uploads its log files, they are saved on the provisioning server with the MAC address of the handset prepended to the file name. For example, **00907a0e360b-boot.log** and **00907a0e360b-app.log** are the files associated with MAC address 00907a0e360b. 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.

Both log files can be uploaded on demand using a multiple key combination described in [Multiple Key Combinations](#). The handset uploads four files, namely, **<mac>-boot.log**, **<mac>-app.log**, **mac-now-boot.log**, and **mac-now-app.log**. The *-now-* logs are uploaded manually unless they are empty. Spectralink 84-Series handset use the 1, 5, and 9 dial pad keys to force log file uploads.

The amount of logging that the handset performs can be tuned for the application to provide more or less detail on specific components of the handset'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 Spectralink Customer Support. Inappropriate logging levels can cause performance issues on the handset.

In addition to logging events, the handset 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 Spectralink Customer Support.

Logging Options

Each of the components of the Spectralink 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 log.cfg template. Log levels range from 0 to 6 (0 for the most detailed (Debug) logging, 6 for the least detailed (fatal errors only). There are many different log types or categories that can be adjusted to assist with the investigation of different problems.

When testing is complete, remember to remove the special troubleshooting configuration parameters from the configuration files.

There are other logging parameters, described 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 handset and the network.

- `log.render.level`—Sets the lowest level that can be logged (default=1) by any of the `log.level.change.module_name` parameters.
- `log.render.file.size`—Maximum size before log file is uploaded (default=32 kb)
- `log.render.file.upload.period`—Frequency, in seconds, of log uploads (default is 86400 seconds = 24 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`—Controls whether to stop or delete logging when the server log reaches its maximum size (default=delete)

Scheduled Logging

Scheduled logging is a powerful tool that can help troubleshoot issues that occur after the handset has been operating for some time.

The output of the logging parameters is written to the application log, and can be examined later (for trend data).

The parameters for scheduled logging are found in the `log.cfg` template. They are `log.sched.x`.

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

scheduled	
log.sched.1.name	showCpuLoad
log.sched.1.level	4
log.sched.1.period	15
log.sched.1.startMode	rel
log.sched.1.startTime	0
log.sched.1.startDay	0
log.sched.2.name	memShow
log.sched.2.level	4
log.sched.2.period	15
log.sched.2.startMode	rel
log.sched.2.startTime	0
log.sched.2.startDay	0

```

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 handset 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 handset (see Multiple Key Combinations – 1 & 5 & 9).

When the log files are manually uploaded, the word *now* is inserted into the name of the file, for example, **00907a0e360b-now-boot.log**.

Logging Modules

Logging Modules include:

app1	httpd	push	tls
brow	ice	sip	utilm
cfg	key	so	wlan
dot1x	pps	tickt	wmgr

Additional log modules can be found in the log.cfg file.

WLAN syslog entries can be very useful in determining AP RSSI, packet retry rates, Network Jitter, Dropped packet rates, Late packets, reasons to roam and many other elements.

Major categories of WLAN entries

- AStats
 - AP MAC address
 - RSSI
 - Payload size, release interval
 - TX packet count

- RX packet count
- Missed packet count
- Dropped packet count
- Jitter in ms
- AThresh
 - AP MAC address
 - RSSI
 - Pay load size, release interval
 - TX packet count
 - RX packet count
 - Missed packet count
 - Dropped packet count
 - Jitter in ms
- NStats
 - AP MAC address
 - AP RSSI
 - TX packet count
 - RX packet count
 - BTX (Broadcast TX) packet count
 - BRX packet count
 - MTX (Multicast TX) packet count
 - MRX packet count
 - TX Drop packet count & %
 - TX Retry packet count & %
 - RX Retry packet count & %
- NThresh
 - AP MAC address
 - AP RSSI
 - TX packet count
 - RX packet count
 - BTX (Broadcast TX) packet count
 - BRX packet count
 - MTX (Multicast TX) packet count
 - MRX packet count
 - TX Drop packet count & %

- TX Retry packet count & %
- RX Retry packet count & %
- Successful Handoff
 - Roam To AP MAC address
 - Roam To channel, Score, RSSI & Penalty
 - Roam From AP MAC address
 - Roam From channel, score, RSSI, Penalty & Reason code
 - Other AP (up to four total) MAC, channel, score, RSSI, penalty & reason code
 - TXPO (Tx power of old AP in dBm)
 - TXPN (TX power of new AP in dBm)
- Failed Handoff
 - Roam TO AP MAC address
 - Roam To channel, score, RSSI, penalty & reason code
 - Previous AP MAC, channel, score, RSSI, penalty & original handoff reason code

This information can be used to evaluate how well the WLAN is supporting the 84-Series handsets.

Low RSSI (too weak) will cause high retries, dropped frames, missed frames, poor audio and poor roaming.



Web Info: Using Syslog on Spectralink handsets

For more information about syslog, see Technical Bulletin CS-14-20: *Syslog on Spectralink Handsets*.

Managing the Phone's Memory Resources

Spectralink handsets 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 handset's memory resources. To ensure your handsets and their configured features operate smoothly, you will need to check that the handsets have adequate available memory resources. If you are using a range of handset features - especially customized or advanced features - you may need to manage handset memory resources. To help you optimize your handset features and memory resources, Spectralink provides several tools and troubleshooting tips.

Identifying Symptoms

When the handset memory resources start to run low, you may notice one or more of the following symptoms:

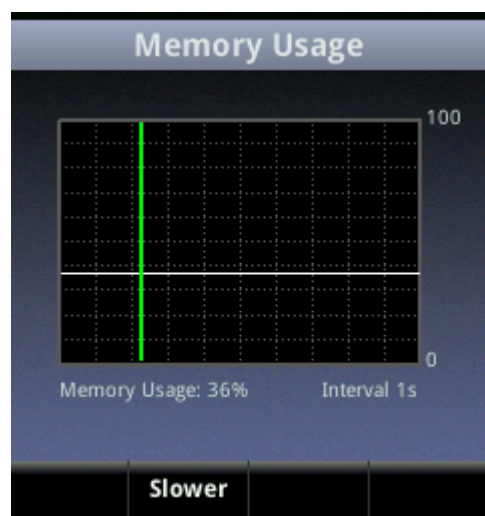
- The handsets reboot or freeze up.
- The handsets do not download all ringtones, directory entries, backgrounds, or XML dictionary files.
- Applications running in the browser stop or do not run at all.

The next sections show you how to check your handset's available memory and manage the handset features to make handset memory available.

Checking the Phone's Available Memory

You can use two methods to quickly check whether you need to manage your handset's memory. Before you begin checking, load and configure the features and files you want to make available on the handset.

Using the first method, on your handset's keypad Select **Settings> 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 handset is running at less than 95 percent of its available memory.

If the handset 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 handset, see [Managing the Phone Features](#).

The second method you can use to confirm whether you need to manage your handset'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 handset prepended to the app log file. For example, if the MAC address of your handset is **00907a0e33b0**, the app log file name will be **00907a0e33b0-app.log**.

Open the app log. If you see the message shown next you may need to manage your handset's memory resources.

Figure 11-6: Application Log Error Message

```

000014.458|dn̄s|*|00|DNS resolver servers are '172.23.0.200' '172.23.0.239'
000014.458|dn̄s|*|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.ld
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 Technical Bulletin CS-14-20: *Syslog on Spectralink Handsets*.

Managing the Phone Features

This section provides tips for managing the handset features to conserve handset 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 Spectralink 84-Series handsets and Polycom desksets, see *Interoperability Guide: Spectralink 84-Series Wireless Telephones and Polycom Desksets*.

All handset features are designed to operate optimally on Spectralink handsets. The features listed in **Table 11-2: Managing the Phone Features** are all customizable, advanced features that can take up significant memory. Use the following table 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 handset features you want for your deployment.

Table 11-2: Managing the Phone Features

Feature	Typical Memory Size
Browser	Variable. Optimized to display three or four elements.
The 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.	
Custom Display Image	15KB
The average size of Spectralink display images is 15KB. If you are using custom images, Spectralink recommends limiting the file size to 15KB for images on the display. If your handset does not display your custom image and the file size is less than 15KB, try reducing the number of available ringtones or display and image backgrounds.	
Local Contact Directory	170 bytes per entry
Spectralink handsets are optimized to display four contact attributes to a maximum of 250 contact entries. Each entry averages about 170 bytes of memory.	

Feature	Typical Memory Size
If you need more space for the contact directory, try reducing the number of available ringtones or image backgrounds.	
Corporate Directory	Varies by server
The Corporate Directory feature is optimized to display five contact attributes up to a maximum of eight. 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 handset has difficulty displaying directory search results with more than five attributes, try reducing the number of available ringtones or image backgrounds, or disable the browser.	
Ringtones	16KB
Spectralink provides a number of audio files for ringtones that are designed to work correctly with the wireless handsets. Spectralink ringtones can range in size from 30KB to 125KB. If you want to use custom ringtones, Spectralink recommends limiting the file size to 16KB. If you want to make more room for custom ringtones, try disabling the 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.	
Background Images	8 – 32KB
Spectralink handsets 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 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.	
Phone Interface Language	90KB
The average size of the XML dictionary files for languages that display on the handset'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, Spectralink recommends using only those XML language files for the languages you need.	
Web Configuration Utility Interface	250KB
The average size of the <i>languages</i> 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, Spectralink recommends using only those XML language files for the languages you need.	

If you are still having difficulty freeing up sufficient space on your handsets, contact Spectralink Voice Product Support.

Testing Phone Hardware

You can view diagnostic information from the **Diagnostics** menu on your handset (**Settings> 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
- **Brightness** – test the brightness of the display
- **LED Diagnostics** – test the LED light on your handset
- **Vibrate** – test the vibrate option

- **Accelerometer** – test using the Personal Alarms application

Uploading a Phone's Configuration

Spectralink Software allows the upload of the files representing a handset'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.

You can upload the handset's configuration from the handset's menu or through the Web Configuration Utility.

This is primarily a diagnostics tool to help find configuration errors.

To upload the handset's current configuration:

- 1 Navigate to the Upload Configuration menu on the handset (**Settings> Advanced Settings> [enter password]> Administration Settings> Upload Configuration**).
- 2 Choose to upload the configuration from one of All Sources, Configuration Files, Local, Web or SIP.
- 3 Press the Upload softkey.

The handset 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

Ping and traceroute are available as diagnostics tools. These diagnostics can be used for troubleshooting network connectivity problems in the wired and wireless networks.

Both tools are accessible by pressing the Home key and left or right arrows to select **Settings> Status> Diagnostics> Network**.

Enter a URL address (for example, `http://www.google.com`) or any IP address (for example, the boot server IP address, SIP server IP address or any other handset's IP address), and then press the **Enter** softkey.

Note: When it is in standby, The 84-Series handset will have ping times which seem large. The handset will go off channel, to scan for other APs, or go to sleep to save battery power. The in-standby handset will not respond as quickly as a wired powered network device. When in call the handset will respond to pings with times often less than 20ms.

Network Protocols and Ports Used on Spectralink Handsets

See the next table for a list of the protocols and ports currently used by the Spectralink Software.

Table 11-3: Protocols & Ports used by Spectralink handsets

Port Number	Protocol	Outgoing	Incoming	UDP or TCP
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
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	
5070	SIP	SIP signaling (Nortel CS1K)	SIP signaling (Nortel CS1K)	
7778	OCS			
14394	QBC Signaling		QBC Server	TCP
24800	PDC	PDC Client messages	PDC Server messages	TCP

¹ Telnet is disabled by default.

² RTP and RTCP can use any set of even/odd ports between 2222 and 2269. This is configurable by setting tcplpApp.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 handset has power issues or the handset has no power.

Determine if the problem is caused by the handset, or the battery. Do one of the following:

- Verify that no lights appear on the unit when it is powered up.
 - Press any key to determine if the display lights up (comes out of battery save mode)
 - Try another battery
 - Try a battery from a working handset
-

The handset will not boot

If your handset will not boot, there may be a corrupt or invalid firmware image or an invalid configuration on the handset:

- Ensure the handset is connected to the WLAN (no red X on the AP signal strength bars)
- Ensure that the provisioning server is accessible on the network and a valid software load and valid configuration files are available.
- Ensure that the handset is pointing to the provisioning server on the network. Check Settings> Status> Platform> Configuration. Boot Server shows IP address. Check to see load protocol configured (FTP/TFTP/HTTP/HTTPS). Check Config for files which load to the handset.
- Reboot the handset.

Battery Pack power specifications

The battery pack voltage measurement triggers the low battery or very low battery alert:

< 3.625 V – Low battery

< 3.500 V – Very low battery (Critically low)

< 3.400 V – Shutdown

Syslog will display battery status with the showBatteryStat tag.

Key Pad Issues

The following table describes possible solutions to issues you may have with the key pad.

Table 11-5: Troubleshooting Key Pad Issues

The key pad does not work

If the dial pad on your handset does not respond, do the following:

- Check for a response from other feature keys or from the key pad.
 - Place a call to the suspect handset from a known working telephone. Check for display updates. Answer the call. Does the handset work properly in call?
 - Press the Settings> Status> Diagnostics> Test Hardware> Keypad Diagnostics> Verify: press a key. Press each key to ensure each key is recognized properly.
 - If the keypad seems defective contact your help desk, the reseller or Spectralink support.
-

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 handset 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 handset to check for inbound call display and ringing. If successful, try to press feature keys while a call is active to access a Directory, for example.
 - Navigate to **Settings> Status> Lines** to confirm the line is actively registered to the call server.
 - Reboot the handset to attempt re-registration to the call server (see [Rebooting the Phone](#)).
-

The display shows the message *Network Link is Down*

- Ping the handset from another machine.
 - Reboot the handset to attempt re-registration to the call server (navigate to **Settings> Basic Settings> Restart Phone**).
-

Calling Issues

The following table provides possible solutions to a number of generic calling issues.

Table 11-7: Troubleshooting Calling Issues

The handset does not ring

If there is a no ring tone, but the handset displays a visual indication when it receives an incoming call, do the following:

- Adjust the ring level 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

If you see unregistered line icon the handset line is unregistered. Register the line and try to place a call.

Unregistered Line Icon:



Registered Line Icon:



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 handset. Do one of the following:

- Check that the display is illuminated. Press any key to wake up the display.
 - Power up the handset. Ensure the battery used is charged.
 - Test the display by **Settings> Status> Diagnostics> Test Hardware> Display Diagnostics**. Watch the test run to see the display change. Press any key to exit the test
 - Use the screen capture feature to determine if the display on the handset is incorrect. See [Capturing the Phone's Current Screen](#).
-

The display is too dark or too light

The handset contrast may be set incorrectly. To adjust the contrast, do one of the following:

- Adjust the backlight intensity. Navigate to **Settings> Basic Settings> Backlight Intensity settings**.
 - Reboot the handset 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 time and date are flashing

If the time and date are flashing, you have disconnected the handset from the LAN or there is no SNTP time server configured. Do one of the following (for instructions, see [Time and Date Display](#)):

- Ensure the handset is connected to the WLAN and to the LAN.
 - Configure an SNTP server.
 - Disable the time and date (if you do not wish to connect your handset 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 headset jack on the handset.
 - Ensure the handset volume is turned up or set to a comfortable volume.
-

There are audio or echo issues

If you experience echo issues, investigate the issue to determine the following:

- Is the problem localized to one 84-Series handset?
 - Are other 84-Series handsets experiencing audio issues?
-

- Is the problem related to one area of the facility?
- Check SIP server settings which may be applicable to the audio or echo issues.
- See Technical Bulletin *Troubleshooting Audio and Echo Issues*

There is choppy/poor audio

Poor audio can be caused by several things including:

- Inadequate RSSI (RF Signaling from the AP)
 - High retry rates
 - High Missed packet rates
 - High Dropped packet rates
 - High Late packet rates
 - Loss of connection to the WLAN
 - Long roam times between APs
 - Congestion on the network between the 84-Series handset, SIP server or other end point
-

Additionally...

Logging data is useful for later syslog data analysis with `render =1` and `WLAN = 1` along with Call start and end times and symptom descriptions at the logging level for WLAN at 2.

Another troubleshooting technique is to use Diagnostics mode to check for many of the above items during the call. Go to **Settings> Status> Diagnostics> WiFi Stats>**

- [1/6] Packet Count to see Missed RX, Missed TX packets, RX Retry count and TX Retry count. This screen shows cumulative packet counts not percentage. If any of these counters are incrementing frequently it is a bad sign.
- [2/6] General Information shows SSID, Last TX rate and TX power
- [3/6] AP List shows last few bytes of the AP MAC, channel and RSSI (a lower number (-45 versus -75 is a stronger signal)
- [4/6] Re/Association Count shows Association count, Reassociation count, Association fail count and Reassociation fail count
- [5/6] Security shows MIC Err Count, MIC Err Last Seq , ICV Err Count and QoS Fail Count
- [6/6] EAP Information shows EAP Err Count, Last EAP Err Code and 802.11n disabled

When in call, focus on screens 1 and 3 to see how packet counts seem to be affected by low (inadequate) RSSI

Upgrading Issues



Tip: Important!

When upgrading handset software using the Web Configuration Utility, the handset is unable to connect to the Spectralink Hosted Server.

Occasionally, the handset is unable to connect to the Spectralink Hosted Server because:

- The Spectralink Hosted Server is temporarily unavailable.
- There isn't any software upgrade file for the handset to download.
- The network configuration is preventing the handset from connecting to the Spectralink Hosted Server. Check for a firewall, ACL or another issue preventing the handset from loading code from the provisioning server.



Note: Cannot upgrade through a Web proxy

Spectralink Software does not support internet access for software upgrades through a Web proxy.

To troubleshoot an upgrade issue:

- Verify that new software is available for your handset.
- Verify that your network's configuration will allow the handset to connect to <http://downloads.Spectralink.com>.
- Check the web site with a computer to verify it is available and the file can download
- Try upgrading the code again
- Use a local provisioning server

If the issue persists, try manually upgrading your handset's software. To upgrade handset software using this method, see [Setting Up the Provisioning Server](#).

Chapter 13: Miscellaneous Maintenance Tasks

This chapter shows you how to maintain the Spectralink Software. This includes:

- Trusted Certificate Authority List
- Encrypting Configuration Files
- Multiple Key Combinations
- Default Feature Key Layouts
- Parsing Vendor ID Information
- Product, Model, and Part Number Mapping
- Capturing the Phone's Current Screen

Encrypting Configuration Files

For security reasons, an administrator may wish to encrypt configuration files sent to the phone from a provisioning server and files uploaded by the phone to the provisioning server. Setting up dynamic encryption involves generating a key, downloading it to the phone, encrypting all config files used by the phone and making these files available to the phone by referencing them in the master config file which itself is not encrypted.



Note: Not all config files can be encrypted

Note that the master configuration file or the bootrom.ld or sip.ld software files cannot be encrypted.

All exe files referred to in this section can be found on the 84-Series webpage under the Downloads tab in the ConfigFileEncryption_v[x].zip. Our config file examples use the configuration templates provided with the software.

Ensure that you securely store all files generated by this procedure.



Admin Tip: Using Cygwin or Unix?

The configuration steps and examples below assume you are using DOS. If you are using Cygwin or Unix, make the appropriate adjustments to the code.

Configuration steps

- 1 Generate a key.



Note: ConfigFileKeyGen.exe requires a random seed file

If using the DOS version of the key generator you can create the “c:\.rnd” and populate it with random data of your choosing.

If compiled under a Linux or Unix system (generally including Cygwin), it should default to /dev/random and there will be no extra steps required. Some installations of Cygwin or Unix do not have a /dev/random device installed. For this situation you can create the .rnd file and populate it with random data of your choosing.

Use the `configFileKeyGen.exe` file to generate the key. To generate a key named “key1.key” use this command format:

```
> configFileKeyGen.exe -k key1.key -d hello
```

The command has these components:

- » -k [the key filename]
- » -d [the key description]

Your generated key (key1) is a string that looks like this:

```
Crypt=1;KeyDesc=hello;Key=93750C896A35F74EF704CEA66CC89049;
```

The string contains three attributes:

- » The type of encryption - 128bit AES encryption [Crypt=1]
- » A description of the key [KeyDesc=hello]
- » The key itself [Key=93750C896A35F74EF704CEA66CC89049]



Admin Tip: Problems with key generation?

Try generating a key using “hello” as the key description as shown in our example.

2 Create an XML .cfg file for the key. Our example is named key1.cfg.

Use the device parameters in the next table:

Table 12-1: Encryption key parameters

Parameter	Permitted Values	Default
device.set	0 or 1	0
Set to 1 to enable encryption		
device.sec.configEncryption.key	[the key]	Null
The key that is generated by the process described in this section. Note that the KeyDesc must be “hello”.		
device.sec.configEncryption.key.set	0 or 1	0
Set to 1 in order to use the key set with <code>device.sec.configEncryption.key</code>		

Example: config file (key1.cfg) for loading the key into the handsets

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<handsetConfig
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="handsetConfig.xsd">

  <device
    device.set="1"
    device.sec.configEncryption.key="Crypt=1;KeyDesc=hello,Key=93750C896A35F74EF704CEA66CC89049;"
    device.sec.configEncryption.key.set="1"/>
  </device>
</handsetConfig>
```



Caution: Use the entire string produced by the key generator

Ensure that you include the full string (including the “Crypt” and “KeyDesc” fields) rather than just the key. Also include the semicolon at the end of the key. The phone will fail to read the key if you omit any part of the string.

- 3 Load the key in the phones. This should be done securely through a USB connection or within a secure wireless lab. The assumption here is that this is part of the initial deployment, all phones are available, config files are developed, and the SLIC configuration is complete.

Whether connected through usb or through the secure LAN, the phone will find the server and look for the master config file. Therefore the key file must be referenced in the master config file, as shown below, and both the key.cfg and master config files must be loaded on the provisioning server.

A master configuration file referencing the key1.cfg file

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<handsetConfig
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="handsetConfig.xsd">

  <MASTER_CONFIG>

    <!-- You can specify a path with subdirectories to specify the -->
    <!-- location of the handset software. -->
    <!-- See the Deployment Guide for more information about setting up-->
    <!--subdirectories.-->

    <CONFIGURATION
      CONFIG_FILES="key1.cfg"
    />

    <DIRECTORIES
      LOG_FILE_DIRECTORY=""
      OVERRIDES_DIRECTORY=""
      CONTACTS_DIRECTORY=""
      CALL_LISTS_DIRECTORY=""
    />

  </MASTER_CONFIG>
```

- a Place the master config file and the key config file onto the provisioning server that the handset is programmed (through SLIC or manually) to access.
- b Plug in (or turn on) the phone and allow it to find and load the file. Do this for all your phones.

Once this is done the phones will be able to decrypt any files that use this key. They will also encrypt files uploaded to the server if you configure it this way. (per step 4) However, they do not have the encrypted config files yet so they will not be able to make calls, etc., just yet.

- c Once the key has loaded to the phone, the key1.cfg file is no longer necessary and the master config file must be edited to include the encrypted config files. For greatest security you can remove these two files from the server and store them securely for later use and configure a new master config file as described in step 6.

You can also comment out the commands or change the device.set parameter to zero. But be aware that the master config file cannot be encrypted.

- 4 (Optional) Add `<sec.encrypted>` parameters to phones' site.cfg config file (or other config file that all encrypted phones will access).

The `<sec.encrypted>` parameters specify whether the phone uploads its configuration files in an encrypted or unencrypted format. See [Configuration File Encryption](#) for exact information.

- 5 Encrypt the config files used by the phones.

Use the `configFileEncrypt.exe` program to encrypt the config files used by the phones.

We recommend that you rename the encrypted file so that it is obvious it is encrypted. For example, here we rename the site.cfg file to site-cfg.enc when we encrypt it using the encryption command. Use the same key name in the command that you just loaded on the phones! This is the command format:

```
> configFileEncrypt.exe -i site.cfg -o site-cfg.enc1 -k key1.key
```

The command has these components:

- » -i [the un-encrypted filename]
- » -o [the encrypted filename]
- » -k [the key filename]

Run `configFileEncrypt.exe` on each config file used by the phones, except of course, the master config file.

Securely store the unencrypted files, just as you stored the key file in step 3c.



Caution: If using a Linux computer to generate the encrypted files

Check the encrypted files to ensure they are not altered when you copy them from a computer running the Linux operating system to a computer running the Microsoft Windows operating system. See [Comparing encrypted and unencrypted files](#).

- 6 Load the encrypted config files into your master configuration file. See the next example.

A master configuration file with encrypted config files referenced:

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<handsetConfig
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="handsetConfig.xsd">

  <MASTER_CONFIG>

    <!-- You can specify a path with subdirectories to specify the -->
    <!-- location of the handset software. -->
    <!-- See the Deployment Guide for more information about setting up-->
    <!--subdirectories.-->

    <CONFIGURATION
      CONFIG_FILES="[PHONE_MAC_ADDRESS]-ext-cfg.enc1, site-cfg.enc1"
    />

    <DIRECTORIES
      LOG_FILE_DIRECTORY=""
      OVERRIDES_DIRECTORY=""
      CONTACTS_DIRECTORY=""
      CALL_LISTS_DIRECTORY=""
    />

  </MASTER_CONFIG>
```

- 7 The phones are now ready to deploy.

Comparing encrypted and unencrypted files

- 8 Run `configFileEncrypt.exe` on the unencrypted file with the "-d" option. This shows the "digest" field. E.g.:

```
> configFileEncrypt.exe -i site.cfg -d
```

 The command has these components:
 - -i [the un-encrypted filename]
 - -d [the digest field request]
- 9 View the `Digest=` line in the resulting file.
- 10 Look at the encrypted file using text editor and check the "`Digest=...`" field.
- 11 If the two fields are the same, then the encrypted and un-encrypted file are the same.

Decrypting existing configuration files

The `configFileDecrypt.exe` program can be used to decrypt an encrypted file as long as the key filename is included in the command. Use the following format:

```
> configFileDecrypt.exe -i site-cfg.enc1 -o site.cfg -k key1.key
```

The command has these components:

- -i [the encrypted filename]
- -o [the decrypted filename]
- -k [the key filename]

Changing an existing key

Changing to a new key is a multi-step process. This is best done by gathering up all the phones and doing them all at once. Phones that are in use will experience a period of disruption as the key is swapped out. Use the following steps:

- 1 Generate a new key following step 1 above and save the file. We'll call it key2.key.
- 2 Create a new .cfg file with the new key. Save it with some way of differentiating it from the previous key config file. Use step 2 above for guidance. We will use key2.cfg.
- 3 Use the old key (key1) to encrypt the key2.cfg file per step 5 above.
- 4 Load the new key into the phone. This time you can do this wirelessly as key2.cfg is already encrypted with key1.
 - a Create a master config file that references this key2.cfg file.
 - b Replace the master config file that the phone has been using with this new one.

The phone will load the key2.cfg file containing the new key by using the old key still on the phone. The phone will then replace the old key (key1) with the new key (key2).
At this point, the phone can no longer read the config files encrypted with key1.
- 5 Use key2 to generate new versions of the encrypted config files. You will need to decrypt each file with the old key and then encrypt each file with the new key. Use a new name for them so that they can easily be differentiated from the files that use the old key. See step 5 above.
- 6 In the master configuration file, replace the configuration files that used the old key with the configuration files that use the new key. Load the master configuration file and all the config files using the new key on the provisioning server.
- 7 Reboot the phones. They should now be able to read the new files.

Log messages

You can look in the app log files for logging related to the configuration files.

If a file cannot be decrypted, messages similar to this will appear in the log:

```
000014.938|so|*|00|Configuration files: 00907a112233-cfg.enc,
pbx_4053.enc 000014.938|so|*|00|Configuration file
"[PHONE_MAC_ADDRESS]-ext-cfg.enc1" SHA1 digest: Unknown
000014.938|so|*|00|Configuration file "site-cfg.enc1" SHA1
digest: Unknown
```


In particular, note the presence of the “Unknown” as the SHA1 digest. Additionally, logs like this may be present:

```
0804163651|cfg |5|00|Prm|Could not decrypt site-cfg.enc1
```

When the config files are properly decrypted, the messages like this should appear:

```
000134.739|so|*|00|Configuration files: [PHONE_MAC_ADDRESS]-ext-
cfg.enc1, site-cfg.enc1 000134.739|so|*|00|Configuration file
"00907a112233-cfg.enc" SHA1 digest:
DB944E56E0413904353A2CD1A0FA29BF69E2AC1E
000134.739|so|*|00|Configuration file " site-cfg.enc1 " SHA1
digest: B4465B48A226DE487445A1519F0D566CDC07BBB3
```

In addition, messages similar to the following may be present, indicating the file was read and parameters were parsed:

```
000134.739|cfg |3|00|Prm|Configuration file(s) statistics: 20
valid parameters found. 000134.739|cfg |3|00|Prm|Configuration
file(s) statistics: 14 parameter values have been used. 6
parameter values were set more than once
```



Troubleshooting: My phone keeps displaying an error message for my encrypted file

If a handset downloads an encrypted file that it cannot decrypt, the action is logged, and an error message displays. The handset 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.

Multiple Key Combinations

You can use multiple key combinations on your Spectralink handsets to reboot the handset, to restore the handset to factory default values, or to upload log files from the handset to your provisioning server.

Rebooting the Phone

Rebooting the handset downloads new software and new configuration files if they exist on the provisioning server.



Timesaver: Download new configuration files without rebooting your phone

Not all configuration parameter changes require the handset to restart or reboot. You can update your handset's configuration by navigating to **Settings> Basic Settings> Update Configuration**. If there is new software (different version) on the provisioning server, the handset will restart or reboot to download the software. If there are configuration file changes, your handset will only restart if it is

necessary. Otherwise, the handset will download the new configuration files without restarting.

You can use a multiple key combination to reboot your handset. Press and hold the 0, 1, and 3 keys simultaneously until you hear a confirmation tone (for about three seconds).



Power Tip: Quickly restarting your phone

Users can restart their handsets by pressing the **Home** key and selecting **Settings> Basic Settings> Restart Phone**. If new Updater or Spectralink Software is available on the provisioning server, the handset will download the software when it restarts.

Resetting to factory defaults

Resetting the handset 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 handset to the factory defaults. Press and hold the 1, 3, and 5 dial pad keys simultaneously during the Updater/BootROM countdown process until the password prompt displays.

Enter the administrator password to initiate the reset. Resetting to factory defaults will also reset the administrator password (factory default password is 456). Spectralink recommends that you change the administrative password from the default value.

Updating log files

Uploading the log files copies the log files from the handset to the provisioning server. The files called **<MACAddress>-now-xxx.log** are created. Xxx is boot or app.

You can use a multiple key press to upload log files to your provisioning server. Press and hold the 1, 5, and 9 dial pad keys simultaneously until you hear a confirmation tone (for about three seconds).

Setting base profile

Setting the base profile allows for quick setup of Spectralink handsets with Microsoft Lync Server 2013 or 2010.

You can use a multiple key combination to set the base profile on a particular Spectralink handset. Depending on your handset model, press and hold the 1, 4, and 9 dial pad keys simultaneously for about three seconds until you hear a confirmation tone.

A login screen displays. Enter the administrator password (default 456) to initiate the setup. Spectralink 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 Spectralink 84-Series wireless handsets.

The illustration of the Spectralink handsets is followed by a table that shows the available handset key functions.

Spectralink 84-Series

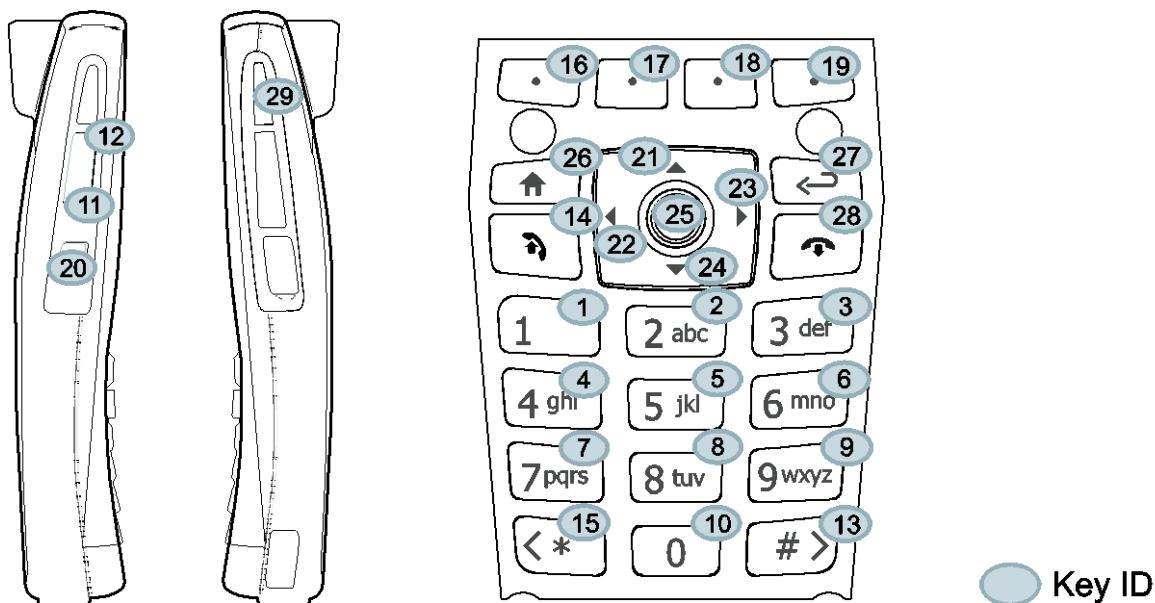


Table 12-2: Spectralink 8440 and 8450 handset key functions

Key ID	Function	Key ID	Function	Key ID	Function
1	Dialpad1	12	VolUp	23	ArrowRight
2	Dialpad2	13	DialpadPound	24	ArrowDown
3	Dialpad3	14	Green	25	Select
4	Dialpad4	15	DialpadStar	26	Home
5	Dialpad5	16	SoftKey1	27	Back
6	Dialpad6	17	SoftKey2	28	Red
7	Dialpad7	18	SoftKey3	29	Barcode
8	Dialpad8	19	SoftKey4		
9	Dialpad 9	20	Talk		
10	Dialpad0	21	ArrowUp		
11	VolDown	22	ArrowLeft		

Parsing Vendor ID Information

After the handset 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 handset's hardware ID and the BootROM version. RFC 2132 does not specify the format of this option's data which 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 Spectralink 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 Spectralink 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 Spectralink signature (always 4 octets)
6f
  o Length of Spectralink data
01 07 50 6f 6c 79 63 6f 6d
  o sub-option 1 (company), length, "Spectralink"
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
```

- sub-option 5 (BootROM version), length, "BR/3.1.0.XXXX/28-Apr-05

```
13:30"
```

```
ff
```

- 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>/<buildid>/<date+time>
```

where:

<apptype> can be 'BR' (BootROM) or 'SIP' (SIP Application)

Product Model Number and Hardware ID Mapping

The master configuration file can direct handset upgrades to a software image and configuration files based on a handset model name, a firmware hardware ID, or a single handset's MAC address.

The hardware ID has precedence over the model name, which has precedence over the default CONFIG_FILES parameter. For example,

```
CONFIG_FILES_3111-36150=001=
```

```
"phone1_3111-36150-001.cfg, sip_3111-36150-001.cfg"
```

will override

```
CONFIG_FILES_SL8450=
```

```
"phone1_SL8450.cfg, sip_SL8450.cfg",
```

which will override

```
CONFIG_FILES=
```

```
"phone1.cfg, sip.cfg"
```

for a Spectralink 8450.

You can also add variables to the master configuration file that are replaced when the handset reboots. The variables include PHONE_MODEL, PHONE_PART_NUMBER, and PHONE_MAC_ADDRESS.

Table 12-3: Model Name and Hardware ID

Model Name	Hardware ID
SL8440	3111-36150-001
SL8450	3111-36152-001
SL8452	3111-36154-001
SL8441	3111-67360-001
SL8453	3111-67361-001

Capturing the Phone's Current Screen

You can capture your handset'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 handset'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:



- 2 Save the configuration file and update your handset's configuration.
- 3 On the handset, turn on the screen capture feature from the **Screen Capture** menu (**Settings> Basic> Preferences> Screen Capture**).
You will need to turn the screen capture on again (repeat this step) each time the handset restarts or reboots.
- 4 In a Web browser, enter `http://<handsetIPaddress>/captureScreen` in the browser address field. (To find your handset's IP address, navigate to **Settings> Status> Platform> Phone**.)
- 5 Enter the username and password, as needed.
- 6 The Web browser will display an image showing the handset's current screen. The image can be saved as a BMP or JPEG file. Right-click the image and save accordingly.
- 7 Refresh your browser to display the current screen on the handset.

Part V: Appendices

Provides reference information about Ringtone Pattern Names and Sound Effects, Session Initiation Protocol (SIP), and information about the third-party software that is included in the 84-Series software.

Appendix A: Ringtone Pattern Names and Sound Effects Parameters

Ringer Patterns

The following table shows the ring pattern names and their default descriptions:

<i>Parameter Name</i>	<i>Ring Type number</i>	<i>Ringtone Name</i>	<i>Description</i>
ringer1	na	Silent Ring	Silent ring
ringer2	1	Low Trill	Long single A3 Db3 major warble
ringer3	2	Low Double Trill	Short double A3 Db3 major warble
ringer4	3	Medium Trill	Long single C3 E3 major warble
ringer5	4	Medium Double Trill	Short double C3 E3 major warble
ringer6	5	High Trill	Long single warble 1
ringer7	6	High Double Trill	Short double warble 1
ringer8	7	Highest Trill	Long single Gb3 A4 major warble
ringer9	8	Highest Double Trill	Short double Gb3 A4 major warble
ringer10	9	Beeble	Short double E3 major
ringer11	10	Triplet	Short triple C3 E3 G3 major ramp
ringer12	11	Ringback-style	Short double ringback
ringer13	12	Low Trill Precedence	Long single A3 Db3 major warble Precedence
ringer14	13	Ring Splash	Splash
ringer15	14	Ring16	Sampled audio file 1
ringer16	15	Ring17	Sampled audio file 2
ringer17	16	Ring18	Sampled audio file 3
ringer18	17	Ring19	Sampled audio file 4
ringer19	18	Ring20	Sampled audio file 5
ringer20	19	Ring21	Sampled audio file 6
ringer21	20	Ring22	Sampled audio file 7
ringer22	21	Ring23	Sampled audio file 8
ringer23	22	Ring24	Sampled audio file 9
ringer24	23	Ring25	Sampled audio file 10



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 handset at the time.



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.

Ring Tones <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

The handset 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.

Sound Effects Ringtone Parameters

Parameter	Permitted Values
se.rt.enabled	0 or 1 (default)
If 0 , the ringtone feature is not enabled on the handset. 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 handset'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 <i>callWaiting</i> .	
se.rt.<ringClass>.name	UTF-8 encoded string
The answer mode for a ringtone. Used for identification purposes in the user interface.	

<i>Parameter</i>	<i>Permitted Values</i>
se.rt.<ringClass>.ringer	default, ringer1 to ringer24
The ringtone to be used for this class of ring. The ringer should match one of the Ringtone Pattern Names in the above table. 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.	

Miscellaneous Patterns

The following table shows the miscellaneous patterns and their descriptions:

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)

Appendix B: Session Initiation Protocol (SIP) Information

This chapter describes the basic Session Initiation Protocol (SIP) and the protocol extensions that the current Spectralink 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.

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

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:

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 handset 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.

Supported SIP Request Headers

<i>Header</i>	<i>Supported</i>
Accept	Yes
Accept-Encoding	Yes
Accept-Language	Yes
Accept-Resource-Priority	Yes

<i>Header</i>	<i>Supported</i>
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
CSeq	Yes
Date	Yes (for missed call, not used to adjust the time of the handset)
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

<i>Header</i>	<i>Supported</i>
Priority	No
Privacy	No
Proxy-Authenticate	Yes
Proxy-Authorization	Yes
Proxy-Require	Yes
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
User-Agent	Yes
Via	Yes
voice-missed-call	Yes
Warning	Yes (Only warning codes 300 to 399)

<i>Header</i>	<i>Supported</i>
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 handset may not actually generate the response.

1xx Responses - Provisional

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

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

Supported 3xx SIP Responses

<i>Response</i>	<i>Supported</i>
300 Multiple Choices	Yes
301 Moved Permanently	Yes

<i>Response</i>	<i>Supported</i>
302 Moved Temporarily	Yes
305 Use Proxy	No
380 Alternative Service	No

4xx Responses - Request Failure



Handling 4xx responses

All 4xx responses for which the handset does not provide specific support will be treated the same as 400 Bad Request.

Supported 4xx SIP Responses

<i>Response</i>	<i>Supported</i>
400 Bad Request	Yes
401 Unauthorized	Yes
402 Payment Required	No
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

<i>Response</i>	<i>Supported</i>
486 Busy Here	Yes
487 Request Terminated	Yes
488 Not Acceptable Here	Yes
491 Request Pending	No
493 Undecipherable	No

5xx Responses - Server Failure

Supported 5xx SIP Responses

<i>Response</i>	<i>Supported</i>
5-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

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 handset 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 handset is configurable using the `voIpProt.SIP.useRFC2543hold` parameter (see [Call Hold](#)), but both methods are supported when signaled by the remote end point.



Hold Methods

Even if the handset 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 handset fully supports RFC 3262 - *Reliability of Provisional Responses*.

Transfer

The handset supports transfer using the REFER method specified in draft-ietf-sip-cc-transfer-05 and RFC 3515.

Third party call control

The handset 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 handset may be used for remote control of the handset from computer applications such as Microsoft Office Communicator.

The handset 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 the SIP parameter table in the [<voIpProt/>](#) section and [Registrations](#) and needs to be activated by a feature application key.

SIP for Instant Messaging and Presence

The handset is compatible with the Presence and Instant Messaging features of Microsoft Windows Messenger 5.1.

Appendix C: Open Source Information

OFFER for Source for GPL and LGPL Software

You may have received a Spectralink 84-Series Wireless Handset from Spectralink that contains—in part—free software (software licensed in a way that allows you the freedom to run, copy, distribute, change, and improve the software).

A complete list of all open source software included in the Spectralink 84-Series Wireless Handset, as well as related license and copyright information, is available at <http://support.spectralink.com>.

You may also obtain the same information by contacting Spectralink by regular mail or email at the addresses listed at the bottom of this notice.

For at least three (3) years from the date of distribution of the applicable product or software, we will give to anyone who contacts us at the contact information provided below, for a charge of no more than our cost of physically distributing, the items listed in “Spectralink OFFER of Source for GPL and LGPL Software” , which is available at <http://support.spectralink.com>.

Contact Information for Requesting Source Code

Spectralink Open Source Manager

2560 55th Street

Boulder, CO 80301

OpenSource@Spectralink.com

Appendix D: Library of <device/> Settings

Spectralink provides a global `device.set` parameter that you can enable for software installation and changes to device parameters.

Device settings are used when the handsets are initially deployed and need to associate with an AP in order to locate the central provisioning server. These are the parameters that are configured by the SLIC tool or when you manually configure the handset for wireless association. Some device parameters are in the `site.cfg` template.

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`.



Caution: Use Caution When Changing Device Parameters

Use caution when changing <device/> parameters as incorrect settings may apply the same IP address to multiple handsets.

Note that some parameters may be ignored. For example, if DHCP is enabled it will still overrule 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 the parameter will not be used.

Incorrect configuration can put the handsets 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, Spectralink recommends that you test the new configuration files on two handsets before initializing all handsets.

The following table outlines the three types of <device/> parameters, their permitted values, and the default value.

Table 0-1: Device Parameters

Parameter	Permitted Values	Default
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.		
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 handset to restart or reboot.

The following table lists each of the `<device/>` parameters that you can configure.

Table 0-2: Device Parameters

Parameter	Permitted Values	Default
device.baseProfile	Generic, Lync	Generic
Choose the Base Profile that the handset will operate with.		
device.dhcp.bootSrvOpt¹	Null, 128 to 254	160
When the boot server is set to <i>Custom</i> or <i>Custom+Option66</i> , specify the numeric DHCP option that the handset will look for.		
device.dhcp.bootSrvOptType¹	IP or String	String
The type of DHCP option in which the handset will look for its provisioning server (if <code>device.dhcp.bootSrvUseOpt</code> is set to <i>Custom</i>). 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	CustomAndDefault
<p>Default The handset 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 handset 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 handset will use the boot server configured through the provisioning server <code>device.prov.*</code> parameters.</p> <p>Custom and Default The handset will use the custom option first or use Option 66 if the custom option is not present.</p>		
device.dhcp.enabled¹	0 or 1	1
If 0, DHCP is disabled. If 1, DHCP is enabled.		
device.dhcp.option60Type¹	Binary, ASCII	ASCII
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.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
device.dhcp.dhcpVlanDiscUseOpt¹	Disabled, Fixed, Custom	Fixed
VLAN Discovery. <i>Disabled</i> , no VLAN discovery through DHCP. <i>Fixed</i> , use predefined DHCP vendor-specific option values of 128, 144, 157 and 191 (<i>device.dhcp.dhcpVlanDiscOpt</i> will be ignored). <i>Custom</i> , use the number specified by <i>device.dhcp.dhcpVlanDiscOpt</i> .		
device.dhcp.dhcpVlanDiscOpt¹	128 to 254	129
The DHCP private option to use when <i>device.dhcp.dhcpVlanDiscUseOpt</i> is set to <i>Custom</i> .		
device.dns.altSrvAddress¹	server address	Null
The secondary server to which the handset directs Domain Name System (DNS) queries.		
device.dns.domain¹	string	Null
The handset's DNS domain.		
device.dns.serverAddress¹	string	Null
The primary server to which the handset directs Domain Name System queries.		
device.hostname¹	string	Null
This parameter enables you to specify a hostname for the handset when using DHCP by adding a hostname string to the handset's configuration. If <i>device.hostname.set=1</i> , and <i>device.hostname=Null</i> , the DHCP client uses Option 12 to send a predefined hostname to the DHCP registration server using <i>Spectralink_<MACaddress></i> . Note that the maximum length of the hostname string is <=255 bytes. The valid character set is defined in RFC1035.		
device.prov.maxRedunServers¹	1 to 8	8
The maximum number of IP addresses that will be used from the DNS.		
device.prov.password¹	string	admin123
The password for the handset to log in to the provisioning server. Note that a password may not be required. <i>Note:</i> If you modify this parameter, the handset will re-provision. The handset may also reboot if the configuration on the provisioning server has changed.		
device.prov.redunAttemptLimit¹	1 to 10	3
The maximum number of attempts to attempt a file transfer before the transfer fails.		
device.prov.redunInterAttemptDelay¹	0 to 300	1
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 (<i>device.dhcp.enabled</i> is 0), if the DHCP server does not send a boot server option, or if the boot server option is static (<i>device.dhcp.bootSrvUseOpt</i> is <i>static</i>). <i>Note:</i> If you modify this parameter, the handset will re-provision. The handset may also reboot if the configuration on the provisioning server has changed.		
device.prov.serverType¹	FTP, TFTP, HTTP, HTTPS, FTPS	FTP
The protocol the handset 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 Upgrade server is an alternate way of getting software updates into the handset through the Web Configuration Utility. It is a completely different process than using a provisioning server method.		

Parameter	Permitted Values	Default
<p>When a value is displayed in this field, it is the address/URL that has been accessed for software updates through the Web Configuration Utility. This value is also stored in the handset's override file on the provisioning server. If this field in the handset menu is populated then you cannot get code onto the phone from any other method than using the WebUI upgrade method because handset settings have highest precedence and this setting is basically a mirror of the override file. If you want to download code into a particular handset using a provisioning server, clear the value set by the WebUI, or edit the override file parameter to "" which will also delete the setting in the handset.</p>		
device.prov.tagSerialNo	0 or 1	0
<p>If 0, the handset's serial number (MAC address) is not included in the User-Agent header of HTTPS/HTTPS transfers and communications to the browser. If 1, the handset's serial number is included.</p>		
device.prov.user	string	administrator
<p>The user name required for the handset to log in to the provisioning server (if required). <i>Note:</i> If you modify this parameter, the handset will re-provision. The handset may also reboot if the configuration on the provisioning server has changed.</p>		
device.sec.configEncryption.key¹	string	Null
<p>The configuration encryption key used to encrypt configuration files. For more information, see Encrypting Configuration Files.</p>		
device.sec.TLS.dot1x.strictCertCommonNameValidation	0 or 1	1
<p>If set to 1, 802.1X always verifies the server certificate for commonName/SubjectAltName match with the server hostname that the phone is trying to connect.</p>		
device.sec.TLS.customCaCert1 (TLS Platform Profile 1) device.sec.TLS.customCaCert2 (TLS Platform Profile 2)	string, PEM format	Null
<p>The custom certificate to use for TLS Platform Profile 1 and TLS Platform Profile 2 device.sec.TLS.profile.caCertList must be configured to use a custom certificate.</p>		
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
<p>Note that you use a single .set parameter to enable or disable only these two related <device/> parameters - device.sec.TLS.customDeviceCertX.publicCert and device.sec.TLS.customDeviceCertX.publicKey. All other <device/> parameters have their own corresponding .set parameter that will enable or disable that parameter.</p>		
device.snmp.gmtOffset	-43200 to 46800	0
<p>The GMT offset – in seconds – to use for daylight savings time, corresponding to -12 to +13 hours.</p>		
device.snmp.serverName	dotted-decimal IP address or domain name string	Null
<p>The SNMP server from which the handset will obtain the current time.</p>		
device.syslog.facility	0 to 23	16
<p>A description of what generated the log message. For more information, see section 4.1.1 or RFC 3164.</p>		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
device.syslog.prependMac¹	0 or 1	0
If 1, the handset's MAC address is pre-pended to the log message sent to the syslog server.		
device.syslog.renderLevel¹	0 to 6	4
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	UDP
The transport protocol that the handset 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	0
If 1, a DHCP Server (which gives out addresses) needs to be started, as opposed to a DHCP Client (which gets an address).		
device.usbnet.ipGateway¹	String	169.254.1.1
The provisioning server IP address.		
device.usbnet.subnetMask¹	String	255.255.0.0
The handset's subnet mask for USBNet.		
device.usbnet.enabled¹	0 or 1	1
If 0, USBNet is disabled. If 1, USBNet is enabled.		
device.usbnet.ipAddress¹	String	169.254.1.2
The handset's dotted-decimal IP address on the USBNet interface.		
device.wifi.noBkgScanRssi	-100 to 0	-100
This parameter allows the handset to scan during standby in order to locate an AP with a stronger signal. As the number gets higher, the handset scans more frequently. At -100 the handset will never scan in standby. The -65 to -75 range will give you periodic scanning and is appropriate for most installations but will need to be tweaked according to the signal quality in the facility. At -45, the phone will scan constantly and battery life will be impaired.		
device.wifi.dhcpEnabled	0 or 1	1
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	1
If 0, 802.11n support is disabled. If 1, 802.11n support is enabled.		
device.wifi.enabled	0 or 1	0
If 0, the wireless interface is disabled. If 1, the wireless interface is enabled.		
device.wifi.ipAddress	String	0.0.0.0
The IP address of the wireless interface (if not using DHCP).		
device.wifi.ipGateway	String	0.0.0.0
The IP gateway address for the wireless interface (if not using DHCP).		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
device.wifi.psk.keyType The key type: key or passphrase.	0 or 1	0
device.wifi.psk.key The hexadecimal key or ASCII passphrase.	string	0xFF
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.		
device.wifi.acMandatory	0 or 1	0
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. When deploying both 84-Series and 87-Series handsets in the same facility using the same Wireless LAN, Wi-Fi Multimedia Admission Control (aka access control, AC or WMM-AC) must be disabled in any handset parameters and APs as it is not supported by 87 Series handsets. Any parameter that requires or enforces AC must be disabled.		
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
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
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 handset matches that of the access point. "7" is also called "Auto" and will use the maximum permitted power setting.		
device.wifi.radio.band5GHzEnable¹	0 or 1	0
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
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
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 handset matches that of the access point. "7" is also called "Auto" and will use the maximum permitted power setting. Note that ETSI regulations limit the maximum setting in Europe. When domain 2 is selected, txPower will not go above 4.		
device.wifi.radio.regulatoryDomain	1, 2, or 10	Null
Available values specify the regulatory domain. The supported values are 1 (North America), 2 (Europe) 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.		
device.wifi.securityMode¹	None, WEP, WPA-PSK, WPA2-PSK, WPA2-Enterprise	None
The wireless security mode.		
device.wifi.ssid¹	String	SSID1
The Service Set Identifier (SSID) of the wireless network.		

<i>Parameter</i>	<i>Permitted Values</i>	<i>Default</i>
device.wifi.subnetMask¹	String	255.0.0.0
The network mask address of the wireless interface (if not using DHCP).		
device.wifi.wep.authType¹	OpenSystem, SharedKey	0
The Wi-Fi WEP authentication type.		
device.wifi.wep.defaultKey¹	1 to 4	1
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
If 0, WEP encryption is disabled. If 1, WEP encryption is enabled.		
device.wifi.wep.keyLength¹	0 or 1	0
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
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
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, EAP-TLS	PEAPv0/MSCHAPv2
The Extensible Authentication Protocol (EAP) to use for 802.1X authentication.		
device.wifi.wpa2Ent.password¹	String	admin123
The WPA2-Enterprise password. The password is used in EAP-PEAP but is not used in EAP-TLS.		
device.wifi.wpa2Ent.roaming¹	OKC, CCKM	OKC
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	administrator
The WPA2-Enterprise user name. Used for EAP-PEAP and EAP-TLS EAP-TLS uses this parameter or the User ID field under Network Interfaces> Wi-Fi Menu> WPA2-Enterprise for the answer to the EAP "Identity" request. For EAP-TLS: The value that is set in the User ID field should match the identity the RADIUS server will accept, which may vary from one RADIUS server to another. If it is necessary for the identity to match the common name from the factory installed certificate, then User ID should be set to the MAC address with lower case letters and no punctuation, for example "00907a0cd9fd".		

¹ Change causes handset to restart or reboot.

Appendix E: Trusted Certificate Authority List

The phone trusts the following certificate authorities by default:

Release 4.13

- Actalis Authentication Root CA
- Starfield Services Root Certificate Authority - G2
- EE Certification Centre Root CA
- Juur-SK
- Atos TrustedRoot 2011
- Autoridad de Certificacion Firmaprofesional CIF A62634068
- Buypass Class 2 CA 1
- Buypass Class 2 Root CA
- Buypass Class 3 Root CA
- CA Disig
- CA Disig Root R1
- CA Disig Root R2
- Chambers of Commerce Root
- Chambers of Commerce Root - 2008
- Global Chambersign Root
- Global Chambersign Root - 2008
- AC RaÃz CerticÃmara S.A.
- Certigna
- Certinomis - AutoritÃ© Racine
- Certinomis - Root CA
- certSIGN ROOT CA
- CFCA EV ROOT
- China Internet Network Information Center EV Certificates Root
- CNNIC ROOT
- ePKI Root Certification Authority
- AAA Certificate Services
- AddTrust Class 1 CA Root
- AddTrust External CA Root
- AddTrust Public CA Root
- AddTrust Qualified CA Root
- COMODO Certification Authority
- COMODO ECC Certification Authority
- COMODO RSA Certification Authority
- Secure Certificate Services
- Trusted Certificate Services

- USERTrust ECC Certification Authority
- USERTrust RSA Certification Authority
- UTN-USERFirst-Client Authentication and Email
- UTN-USERFirst-Hardware
- UTN-USERFirst-Object
- ComSign CA
- ComSign Secured CA
- EC-ACC
- SecureSign RootCA11
- D-TRUST Root Class 3 CA 2 2009
- D-TRUST Root Class 3 CA 2 EV 2009
- S-TRUST Authentication and Encryption Root CA 2005:PN
- S-TRUST Universal Root CA
- TC TrustCenter Class 3 CA II
- Baltimore CyberTrust Root
- Cybertrust Global Root
- DigiCert Assured ID Root CA
- DigiCert Assured ID Root G2
- DigiCert Assured ID Root G3
- DigiCert Global Root CA
- DigiCert Global Root G2
- DigiCert Global Root G3
- DigiCert High Assurance EV Root CA
- DigiCert Trusted Root G4
- Class 2 Primary CA
- E-Tugra Certification Authority
- EBG Elektronik Sertifika Hizmet Sağlayıcısı
- ACEDICOM Root
- Entrust Root Certification Authority
- Entrust Root Certification Authority - EC1
- Entrust Root Certification Authority - G2
- Entrust.net Certification Authority (2048)
- GlobalSign
- GlobalSign
- GlobalSign ECC Root CA - R4
- GlobalSign ECC Root CA - R5
- GlobalSign Root CA
- Go Daddy Class 2 CA
- Go Daddy Root Certificate Authority - G2
- Starfield Class 2 CA
- Starfield Root Certificate Authority - G2
- IGC/A
- Hongkong Post Root CA 1

- ApplicationCA - Japanese Government
- ACCVRAIZ1
- Root CA Generalitat Valenciana
- Taiwan GRCA
- Staat der Nederlanden EV Root CA
- Staat der Nederlanden Root CA
- Staat der Nederlanden Root CA - G2
- Staat der Nederlanden Root CA - G3
- T  B  TAK UEKAE K  k Sertifika Hizmet Sa  lay  c  s   - S  r  m 3
- Hellenic Academic and Research Institutions RootCA 2011
- DST ACES CA X6
- DST Root CA X3
- IdenTrust Commercial Root CA 1
- IdenTrust Public Sector Root CA 1
- Izenpe.com
- Microsec e-Szigno Root CA
- Microsec e-Szigno Root CA 2009
- NetLock Arany (Class Gold) F  tan  s  tv  ny
- NetLock Expressz (Class C) Tanusitvanykiado
- NetLock Kozjegyzoi (Class A) Tanusitvanykiado
- NetLock Minositett Kozjegyzoi (Class QA) Tanusitvanykiado
- NetLock Uzleti (Class B) Tanusitvanykiado
- PSCProcert
- QuoVadis Root CA 1 G3
- QuoVadis Root CA 2
- QuoVadis Root CA 2 G3
- QuoVadis Root CA 3
- QuoVadis Root CA 3 G3
- QuoVadis Root Certification Authority
- RSA Security 2048 v3
- Security Communication EV RootCA1
- Security Communication Root CA
- Security Communication RootCA2
- StartCom Certification Authority
- StartCom Certification Authority
- StartCom Certification Authority G2
- Swisscom Root CA 1
- Swisscom Root CA 2
- Swisscom Root EV CA 2
- SwissSign Gold CA - G2
- SwissSign Platinum CA - G2
- SwissSign Silver CA - G2
- Equifax Secure CA

- Equifax Secure eBusiness CA-1
- Equifax Secure Global eBusiness CA-1
- GeoTrust Global CA
- GeoTrust Global CA 2
- GeoTrust Primary Certification Authority
- GeoTrust Primary Certification Authority - G2
- GeoTrust Primary Certification Authority - G3
- GeoTrust Universal CA
- GeoTrust Universal CA 2
- thawte Primary Root CA
- thawte Primary Root CA - G2
- thawte Primary Root CA - G3
- VeriSign Class 1 Public PCA
- VeriSign Class 1 Public PCA - G3
- VeriSign Class 1 Public PCA â€™ G2
- VeriSign Class 2 Public PCA - G3
- VeriSign Class 2 Public PCA â€™ G2
- VeriSign Class 3 Public PCA
- VeriSign Class 3 Public PCA - MD2
- VeriSign Class 3 Public PCA â€™ G2
- VeriSign Class 3 Public Primary Certification Authority - G3
- VeriSign Class 3 Public Primary Certification Authority - G4
- VeriSign Class 3 Public Primary Certification Authority - G5
- VeriSign Universal Root Certification Authority
- Deutsche Telekom Root CA 2
- T-TeleSec GlobalRoot Class 2
- T-TeleSec GlobalRoot Class 3
- TWCA Global Root CA
- TWCA Root Certification Authority
- Sonera Class1 CA
- Sonera Class2 CA
- TeliaSonera Root CA v1
- AffirmTrust Commercial
- AffirmTrust Networking
- AffirmTrust Premium
- AffirmTrust Premium ECC
- Trustis FPS Root CA
- Secure Global CA
- SecureTrust CA
- XRamp Global Certification Authority
- TÃœRKTRUST Elektronik Sertifika Hizmet SaÃŸlayÃ±cÃ±sÃ±
- TÃœRKTRUST Elektronik Sertifika Hizmet SaÃŸlayÃ±cÃ±sÃ± H5
- TÃœRKTRUST Elektronik Sertifika Hizmet SaÃŸlayÃ±cÃ±sÃ± H6

- Certum CA
- Certum Trusted Network CA
- Visa eCommerce Root
- Network Solutions Certificate Authority
- WellsSecure Public Root Certificate Authority
- OISTE WISeKey Global Root GA CA
- OISTE WISeKey Global Root GB CA
- CA WoSign ECC Root
- CA æ²fë€šæ¹è• ä¹¹
- Certification Authority of WoSign
- Certification Authority of WoSign G2

Release 4.12

- ACCVRAIZ1
- ACEDICOM Root
- Actalis Authentication Root CA
- AddTrust External Root
- AddTrust Low-Value Services Root
- AddTrust Public Services Root
- AddTrust Qualified Certificates Root
- AffirmTrust Commercial
- AffirmTrust Networking
- AffirmTrust Premium
- AffirmTrust Premium ECC
- ApplicationCA - Japanese Government
- Atos TrustedRoot 2011
- A-Trust-nQual-03
- Autoridad de Certificacion Firmaprofesional CIF A62634068
- Baltimore CyberTrust Root
- Buypass Class 2 CA 1
- Buypass Class 2 Root CA
- Buypass Class 3 CA 1
- Buypass Class 3 Root CA
- CA Disig
- CA Disig Root R1
- CA Disig Root R2
- Camerfirma Chambers of Commerce Root
- Camerfirma Global Chambersign Root
- Certigna
- Certinomis - AutoritÃ© Racine
- Certplus Class 2 Primary CA
- certSIGN ROOT CA

- Certum Root CA
- Certum Trusted Network CA
- CFCA EV ROOT
- Chambers of Commerce Root - 2008
- China Internet Network Information Center EV Certificates Root
- CNNIC ROOT
- Comodo AAA Services root
- COMODO Certification Authority
- COMODO ECC Certification Authority
- COMODO RSA Certification Authority
- Comodo Secure Services root
- Comodo Trusted Services root
- ComSign Secured CA
- Cybertrust Global Root
- Deutsche Telekom Root CA 2
- DigiCert Assured ID Root CA
- DigiCert Assured ID Root G2
- DigiCert Assured ID Root G3
- DigiCert Global Root CA
- DigiCert Global Root G2
- DigiCert Global Root G3
- DigiCert High Assurance EV Root CA
- DigiCert Trusted Root G4
- DST ACES CA X6
- DST Root CA X3
- D-TRUST Root Class 3 CA 2 2009
- D-TRUST Root Class 3 CA 2 EV 2009
- EBG Elektronik Sertifika Hizmet Sağlama Kurumu Root CA
- EC-ACC
- EE Certification Centre Root CA
- Entrust Root Certification Authority
- Entrust Root Certification Authority - EC1
- Entrust Root Certification Authority - G2
- Entrust.net Premium 2048 Secure Server CA
- ePKI Root Certification Authority
- Equifax Secure CA
- E-Tugra Certification Authority
- GeoTrust Global CA
- GeoTrust Global CA 2
- GeoTrust Primary Certification Authority
- GeoTrust Primary Certification Authority - G2
- GeoTrust Primary Certification Authority - G3
- GeoTrust Universal CA

- GeoTrust Universal CA 2
- Global Chambersign Root - 2008
- GlobalSign ECC Root CA - R4
- GlobalSign ECC Root CA - R5
- GlobalSign Root CA
- GlobalSign Root CA - R2
- GlobalSign Root CA - R3
- Go Daddy Class 2 CA
- Go Daddy Root Certificate Authority - G2
- Hellenic Academic and Research Institutions RootCA 2011
- Hongkong Post Root CA 1
- IdenTrust Commercial Root CA 1
- IdenTrust Public Sector Root CA 1
- IGC/A
- Izenpe.com
- Juur-SK
- Microsec e-Szigno Root CA
- Microsec e-Szigno Root CA 2009
- NetLock Arany (Class Gold) Főtanácsátváltiny
- NetLock Notary (Class A) Root
- Network Solutions Certificate Authority
- OISTE WISEKey Global Root GA CA
- PSCProcert
- QuoVadis Root CA
- QuoVadis Root CA 1 G3
- QuoVadis Root CA 2
- QuoVadis Root CA 2 G3
- QuoVadis Root CA 3
- QuoVadis Root CA 3 G3
- Root CA Generalitat Valenciana
- RSA Security 2048 v3
- Secure Global CA
- SecureSign RootCA11
- SecureTrust CA
- Security Communication EV RootCA1
- Security Communication Root CA
- Security Communication RootCA2
- Sonera Class 2 Root CA
- Staat der Nederlanden EV Root CA
- Staat der Nederlanden Root CA
- Staat der Nederlanden Root CA - G2
- Staat der Nederlanden Root CA - G3
- Starfield Class 2 CA

- Starfield Root Certificate Authority - G2
- Starfield Services Root Certificate Authority - G2
- StartCom Certification Authority
- StartCom Certification Authority
- StartCom Certification Authority G2
- Swisscom Root CA 1
- Swisscom Root CA 2
- Swisscom Root EV CA 2
- SwissSign Gold CA - G2
- SwissSign Silver CA - G2
- T\xc3\x9c\x42\xC4\xB0TAK UEKAE K\xC3\xB6k Sertifika Hizmet
Sa\xC4\x9Flay\xC4\xB1\x63\xC4\xB1s\xC4\xB1 - S\xC3\xBCr\xC3\xBCm 3
- Taiwan GRCA
- TC TrustCenter Class 2 CA II
- TC TrustCenter Universal CA I
- TeliaSonera Root CA v1
- thawte Primary Root CA
- thawte Primary Root CA - G2
- thawte Primary Root CA - G3
- Trustis FPS Root CA
- T-TeleSec GlobalRoot Class 2
- T-TeleSec GlobalRoot Class 3
- TURKTRUST Certificate Services Provider Root 1
- TURKTRUST Certificate Services Provider Root 2
- TURKTRUST Certificate Services Provider Root 2007
- TWCA Global Root CA
- TWCA Root Certification Authority
- USERTrust ECC Certification Authority
- USERTrust RSA Certification Authority
- UTN DATACorp SGC Root CA
- UTN USERFirst Hardware Root CA
- Verisign Class 3 Public Primary Certification Authority - G3
- Verisign Class 3 Public Primary Certification Authority - G4
- Verisign Class 3 Public Primary Certification Authority - G5
- Verisign Class 4 Public Primary Certification Authority - G3
- Verisign Universal Root Certification Authority
- Visa eCommerce Root
- WellsSecure Public Root Certificate Authority
- WoSign
- WoSign China
- XRamp Global CA Root

Pre-release 4.12

- 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

Spectralink 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, contact Spectralink Support for help. 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 Spectralink handsets \(Technical Bulletin 17877\)*](#).

Appendix F: Spectralink Certificates

Spectralink CA certificates can be obtained from:

<http://pki.spectralink.com/aia/Spectralink%20Issuing%20CA.crt>

<http://pki.spectralink.com/aia/Spectralink%20Root%20CA.crt>

END OF DOCUMENT