Administrator Guide

SoundPoint®/SoundStation® IP SIP

Version 1.6.x

18 July 2005

Notices

1. Specifications subject to change without notice.

Polycom, Inc.

1565 Barber Lane, Milpitas CA 95035, USA

www.polycom.com

Part Number: 1725-11530-160 Rev A

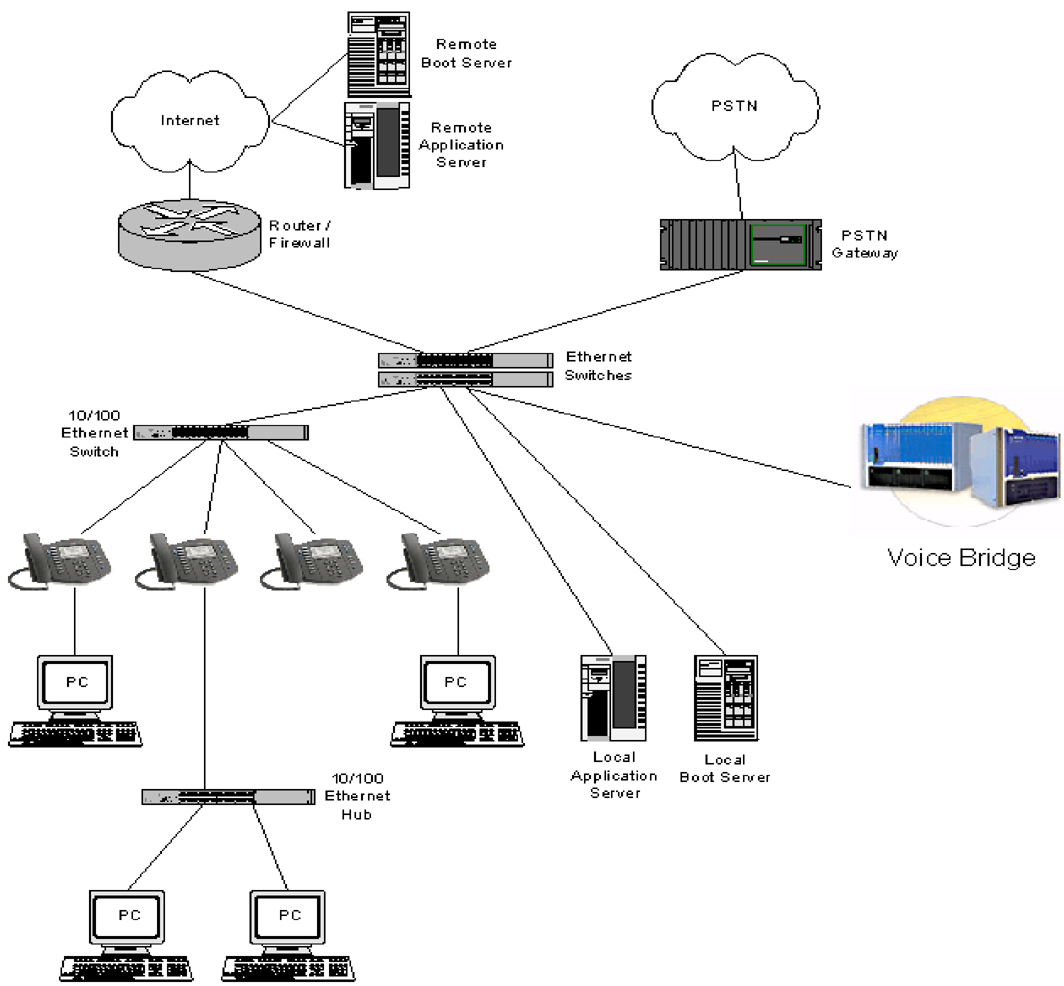
Table of Contents

1 Overview

This Administrator Guide is for the SIP 1.6.0 software release, and the bootROM 3.1.0 release.

Unless specifically described separately,
the behavior and configuration of the SoundPoint® IP 301 is the same as the 300,
the behavior and configuration of the SoundPoint® IP 501 is the same as the 500,
the behavior and configuration of the SoundPoint® IP 601 is the same as the 600.

SoundPoint® IP and SoundStation® IP are feature-rich, enterprise-class voice communications terminals for Ethernet TCP/IP networks. They are designed to facilitate high-quality audio and text message communications. These phones are endpoints in the overall network topology designed to interoperate with other compatible equipment including application servers, media servers, internetworking gateways, voice bridges, and other endpoints.



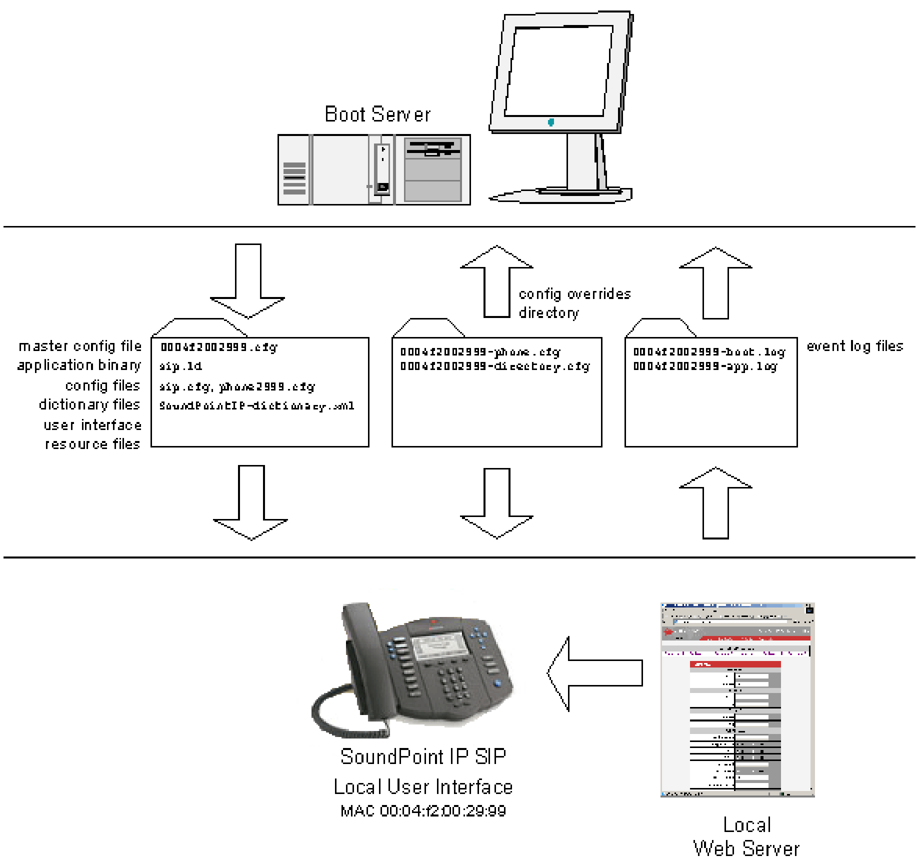
The phones connect physically to a standard office twisted-pair (IEEE 802.3) 10/100 megabytes per second Ethernet LAN and send and receive all data using the same packet-based technology. Since the phone is a data terminal, digitized audio being just another type of data from its perspective, the phone is capable of vastly more than traditional business phones. As SoundPoint® IP and SoundStation® IP run the same protocols as your office personal computer, many innovative applications can be developed without resorting to specialized technology. Regardless of the diverse application potential, it is fundamentally a good office phone, providing the productivity enhancing features needed today such as multiple call appearances, full-duplex speakerphone, hold, transfer, conference, forward, voice mail compatibility, and contact directory.

2 Installation and Operation

This section describes the basic steps that are needed to make your phone operational.

2.1 Installation Models

There are diverse installation models scaling from stand-alone phones to large, centrally provisioned systems with thousands of phones. For any size system, the phones can be centrally provisioned from a boot server via a system of global and per-phone configuration files. To augment the central provisioning model, or as the sole method in smaller systems, configuration can be done using user interfaces driven from the phones themselves: both a local setup user interface and a web server-based user interface are available to make configuration changes.



A boot server allows global and per-phone configuration to be managed centrally via text XML-format configuration files that are downloaded by the phones at boot time. The boot server also facilitates automated application upgrades, diagnostics, and a measure of fault tolerance.

The configuration served by the boot server can be augmented by changes made locally on the phone itself or via the phone's built-in web server. If file uploads are permitted, the boot server allows these local changes to be backed up automatically.

Polycom recommends the boot server central provisioning model for installations involving more than a few phones. The investment required is minimal in terms of time and equipment, and the benefits are significant.

The phones also support dynamic host configuration protocol (DHCP). When set up, DHCP permits plug-and-play TCP/IP network setup.

2.2 Installation Process

Regardless of whether or not you will be installing a centrally provisioned system, there are two steps required to get your phones up and running.

1. Basic TCP/IP Network Setup such as IP address and subnet mask. For more information, see 2.2.1 Basic Network Setup on page 4.
2. Application Configuration such as application specific parameters. For more information, see 2.2.2 Application Configuration on page 11.

2.2.1 Basic Network Setup

The phones boot up in two phases:

- Phase 1: bootROM - a generic program designed to load the application.
- Phase 2: application - the SIP phone application.

Networking starts in Phase 1. The bootROM application uses the network to query the boot server for upgrades or configuration changes, which is an optional process that will happen automatically when properly deployed. The boot server can be on the local LAN or anywhere on the Internet. The bootROM then loads the configured application. The application will restart networking using most of the parameters established by the bootROM (a DHCP query will be performed by the application).

Basic network settings can be changed during Phase 1 using the bootROM's setup menu. A similar, but more sophisticated menu system is present in the application for changing the same network parameters. For more information, see 2.2.1.3 Local User Interface Setup Menus on page 7.

2.2.1.1 DHCP or Manual TCP/IP Setup

Basic network settings can be derived from DHCP or entered manually using the phone's LCD-based user interface. Polycom recommends using DHCP where possible to eliminate repetitive manual data entry.

The following table shows the manually entered networking parameters that may be overridden by parameters obtained from a DHCP server:

Parameter	DHCP Option	DHCP	Configuration File (Phase 2: application only)	Local FLASH
		➪ priority when more than one source exists ➪		
		1	2	3
IP address	1	•	-	•
subnet mask	1	•	-	•
IP gateway	3	•	-	•
boot server address	See 2.2.1.3.2 DHCP Menu on page 8	•	-	•
SNTP server address	42 then 4	•	•	•
SNTP GMT offset	2	•	•	•
DNS server IP address	6	•	-	•
alternate DNS server IP address	6	•	-	•
DNS domain	15	•	-	•
VLAN ID	See 2.2.1.3.2 DHCP Menu on page 8	**Special Case:** Cisco Discovery Protocol (CDP)[a] overrides Local FLASH which overrides DHCP VLAN Discovery.		

a. Can be obtained from a connected Ethernet switch if the switch supports CDP.

2.2.1.2 Provisioning File Transfer

The bootROM on the phone performs the provisioning functions of downloading the bootROM, the <*Ethernet address*>.cfg file, and the SIP application and uploading log files. The SIP application performs the provisioning functions of downloading all other configuration files, uploading and downloading the configuration override file and user directory, downloading the dictionary and uploading log files.

The protocol which will be used to transfer files from the boot server depends on several factors including the phone model and whether the bootROM or SIP application stage of provisioning is in progress. TFTP and FTP are supported by all SoundPoint® and SoundStation® phones. The SoundPoint® IP 301, 501, 600 and 601 and SoundStation® IP 4000 bootROM also supports HTTP while the SIP application supports HTTP[1] and HTTPS. If an unsupported protocol is specified, this may result in unexpected behavior, see the table for details of which protocol the phone will use. The "Specified Protocol" listed in the table can be selected in the Server Type field or the Server Address can include a transfer protocol, for example http://usr:pwd@server (see 2.2.1.3.3 Server Menu on page 10). The boot server address can also be obtained via DHCP. Configuration file names in the <*Ethernet address*>.cfg file can include a transfer protocol, for example https://usr:pwd@server/dir/file.cfg. If a user name and password are specified as part of the server address or file name, they will be used only if the server supports them.

URL Notes: A URL should contain forward slashes instead of back slashes and should not contain spaces. Escape characters are not supported. If a user name and password are not specified, the Server User and Server Password will be used (see 2.2.1.3.3 Server Menu on page 10).

Specified Protocol	Protocol used by bootROM		Protocol used by SIP Application	
	300, 500	301, 501, 600, 601, 4000	300, 500	301, 501, 600, 601, 4000
FTP	FTP	FTP	FTP	FTP
TFTP	TFTP	TFTP	TFTP	TFTP
HTTP	**FTP**	HTTP	HTTP	HTTP
HTTPS	**FTP**	**HTTP**	Not supported. Transfers will fail.	HTTPS

For downloading the bootROM and application images to the phone, the secure HTTPS protocol is not available. To guarantee software integrity, the bootROM will only download signed bootROM or application images. For HTTPS, widely recognized certificate authorities are trusted by the phone and custom certificates can be added. See 6.1 Trusted Certificate Authority List on page 151. Using HTTPS requires that SNTP be functional. Provisioning of configuration files is done by the application instead of the bootROM and this transfer can use a secure protocol.

1. HTTP is supported on all phones to download ringer wave files.

2.2.1.3 Local User Interface Setup Menus

Access to Network Configuration Menu:	
Phase 1: bootROM	The network configuration menu is accessible during the auto-boot countdown of the bootROM phase of operation. Press the **SETUP** soft key to launch the main menu.
Phase 2: application	The network configuration menu is accessible from the main menu. Navigate to Menu>Settings>Advanced>Admin Settings>Network Configuration. Advanced Settings locked by default. Enter the administrator password to unlock. (Factory default password: 456)

Phone network configuration parameters may be edited by means of a main menu and two sub-menus: DHCP Menu and Server Menu.

Use the soft keys, the arrow keys, the *Sel/✓*, and the *Del/X* keys to make changes.

Parameters that cannot be changed are read-only due to the value of other parameters. For example, if the DHCP Client parameter is enabled, the Phone IP Addr and Subnet Mask parameters are dimmed or not visible since these are guaranteed to be supplied by the DHCP server (mandatory DHCP parameters) and the statically assigned IP address and subnet mask will never be used in this configuration.

2.2.1.3.1 Main Menu

Configuration parameters that may be edited on the main setup menu are described in the table below:

Name	Possible Values[a]	Description
DHCP Client	Enabled, Disabled	If enabled, DHCP will be used to obtain the parameters discussed in 2.2.1.1 DHCP or Manual TCP/IP Setup on page 5.
DHCP Menu		See 2.2.1.3.2 DHCP Menu on page 8. Note: Disabled when DHCP client is disabled.
Phone IP Address	dotted-decimal IP address	Phone's IP address. Note: Disabled when DHCP client is enabled.
Subnet Mask	dotted-decimal subnet mask	Phone's subnet mask. Note: Disabled when DHCP client is enabled.
IP Gateway	dotted-decimal IP address	Phone's default router.
Server Menu		See 2.2.1.3.3 Server Menu on page 10.

Name	Possible Values[a]	Description
SNTP Address	dotted-decimal IP address OR domain name string	SNTP server from which the phone will obtain the current time.
GMT Offset	-12 through +13	Offset of the local time zone from Greenwich Mean Time in half hour increments.
DNS Server	dotted-decimal IP address	Primary server to which the phone directs Domain Name System queries.
DNS Alternate Server	dotted-decimal IP address	Secondary server to which the phone directs Domain Name System queries.
DNS Domain	domain name string	Phone's DNS domain.
CDP	Enabled, Disabled	If enabled, the phone will attempt to determine its VLAN ID via the CDP.
VLAN ID	Null, 0 through 4095	Phone's 802.1Q VLAN identifier. Note: 4095 = no VLAN tagging

a. A parameter value of "???" indicates that the parameter has not yet been set and saved in the phone's configuration. Any such parameter should have its value set before continuing.

The DHCP and Server sub-menus may be accessed from the main setup menu.

2.2.1.3.2 DHCP Menu

The DHCP menu is accessible only when the DHCP client is enabled. DHCP configuration parameters are described in the following table:

Name	Possible Values	Description
Timeout	1 through 600	Number of seconds the phone waits for secondary DHCP Offer messages before selecting an offer.

Name	Possible Values	Description
Boot Server	Option 66 Custom Static Custom+Opt.66	Option 66: The phone will look for option number 66 (string type) in the response received from the DHCP server. The DHCP server should send address information in option 66 which matches one of the formats described for Server Address in 2.2.1.3.3 Server Menu on page 10. If the DHCP server sends nothing then the boot server address from flash will be used. Custom: The phone will look for the option number specified by the “Boot Server Option” parameter (below), and the type specified by the “Boot Server Option Type” parameter (below) in the response received from the DHCP server. Static: The phone will use the boot server configured via the Server Menu. For more information, see 2.2.1.3.3 Server Menu on page 10. Custom+Opt.66: The phone will first use the custom option if present or use Option 66 if the custom option is not present.
Boot Server Option	128 through 254 (Cannot be the same as VLAN ID Option)	When the boot server parameter is set to Custom, this parameter specifies the DHCP option number in which the phone will look for its boot server.
Boot Server Option Type	IP Address String	When the Boot Server parameter is set to Custom, this parameter specifies the type of the DHCP option in which the phone will look for its boot server. The IP Address must specify the boot server. The String must match one of the formats described for Server Address in 2.2.1.3.3 Server Menu on page 10
VLAN Discovery	Disabled Fixed Custom	No VLAN discovery via DHCP. Use predefined DHCP private option values of 128, 144, 157 and 191. If this is used, the VLAN ID Option field will be ignored. Use the number specified in the VLAN ID Option field as the DHCP private option value.
VLAN ID Option	128 through 254 (Cannot be the same as Boot Server Option)	The DHCP private option value (when VLAN Discovery is set to Custom). Default is 129.

2.2.1.3.3 Server Menu

Name	Possible Values	Description
Server Type	FTP or Trivial FTP or HTTP or HTTPS	The protocol which the phone will use to obtain configuration and phone application files from the boot server. See 2.2.1.2 Provisioning File Transfer on page 5. FTP = File Transfer Protocol Trivial FTP = Trivial File Transfer Protocol HTTP = Hypertext Transfer Protocol HTTPS = Hypertext Transfer Protocol, Secure
Server Address	dotted-decimal IP address OR domain name string OR URL. All addresses can be followed by an optional directory and optional file name.	The boot server to use if the DHCP client is disabled, or the DHCP server does not send a boot server option, or the Boot Server parameter is set to Static. If a URL is chosen it can include a user name and password. See 2.2.1.2 Provisioning File Transfer on page 5. All options can specify a directory and the master configuration file. See 2.2.2.1.1.1 Master Configuration Files on page 12. Note: ":", "@", or "/" cannot be used in the user name or password.
Server User	any string	The user name used when the phone logs into the server if required for the selected Server Type. Note: If the Server Address is a URL with a user name, this will be ignored.
Server Password[a]	any string	The password used when the phone logs in to the server if required for the selected Server Type. Note: If the Server Address is a URL with user name and password, this will be ignored.
Provisioning Method[b]	Default or SAS-VP v2	If SAS-VP v2 is selected, provisioning is done using XML post/response transactions.
Provisioning String	any string	The string used in XML post/response transactions. Note: Disabled when Provisioning Method is Default.

a. The server user name and password should be changed from the default values. Note that for insecure protocols the user chosen should have very few privileges on the server.

b. Not available on SoundPoint® IP 300 and SoundPoint® IP 500 phones.

2.2.1.4 Reset to Factory Defaults

The basic network configuration referred to in the preceding sections can be reset to factory defaults. To perform this function on all phones except the IP® 4000, simultaneously press and hold the 4, 6, 8 and * dial pad keys until the password prompt appears. To perform this function on the IP® 4000, simultaneously press and hold the 6, 8 and * dial pad keys until the password prompt appears. Enter the administrator password to initiate the reset. ***This will reset the administrator password as well.***

2.2.2 Application Configuration

While it is possible to make calls with the phone using its default configuration, most installations will require some basic configuration changes to get things running optimally. These changes can be made using the central boot server model, if a boot server has been set up, or some, but not all changes can be made using the phone's internal configuration web server or the phone's SIP Configuration menu.

Advantages of using a boot server:

1. The centralized repository for application images and configuration files permits application updates and coordinated configuration parameters.
2. Some parameters can only be modified using boot server configuration files.
3. The multilingual feature requires boot server-resident dictionary files.
4. The customized sound effect wave files require a boot server.
5. When file uploads are permitted, the boot server is the repository for:
   - boot process and application event log files - very effective when diagnosing system problems,
   - local configuration changes via the <*Ethernet address*>-phone.cfg boot server configuration overrides file - the phone treats the boot server copy as the original when booting,
   - per-phone contact directory named <*Ethernet address*>-directory.cfg.
6. The boot server copy of the application images and configuration files can be used to "repair" a damaged phone configuration in the same way that system repair disks work for PCs.

The following sections discuss the available configuration options.

2.2.2.1 Centralized Configuration

The phone application consists of an executable image file (sip.ld) and one or more XML-format configuration files. In the centrally provisioned model, these files are stored on a boot server and cached in the phone. If the boot server is available at boot time, the phone will automatically synchronize its configuration cache with the boot server: bootROM image, application executable, and configuration files are all upgraded this way.

2.2.2.1.1 Configuration Files

The phone configuration files consist of *master configuration files* and *application configuration files*.

2.2.2.1.1.1 Master Configuration Files

Central provisioning requires that an XML-format *master configuration file* be located on the boot server. Either a URL-specified master configuration file or one whose name is associated with the particular phone can be used. Refer to the following sections.

Specified Master Configuration File

The master configuration file can be explicitly specified in the boot server address, for example, http://usr:pwd@server/dir/example1.cfg. The file name must end with ".cfg" and be at least five characters long. If this file cannot be downloaded, the phone will search for the per-phone master configuration file described below.

Per-phone Master Configuration File

If per-phone customization is required (for all applications that require per-phone customization), the file should be named <*Ethernet address*>.cfg, where *Ethernet address* is the Ethernet MAC address of the phone in question. For A-F hexadecimal digits, use lower case only, for example, 0004f200106c.cfg. The Ethernet address can be viewed using the **ABOUT** soft key during the auto-boot countdown of the bootROM or via the Menu>Status>Platform>Phone menu in the application. It is also printed on a label on the back of the phone. If this file cannot be downloaded, the phone will search for the default master configuration file described below.

Default Master Configuration File

For systems in which the configuration is identical for all phones (no per-phone <*Ethernet address*>.cfg files), the default master configuration file may be used to set the configuration for all phones. The file named 000000000000.cfg (<12 zeros>.cfg) is the default master configuration file and it is recommended that one be present on the boot server. If a phone does not find its own <*Ethernet address*>.cfg file, it will use this one, and establish a baseline configuration. This file is part of the standard Polycom distribution of configuration files. It should be used as the template for the <*Ethernet address*>.cfg files.

The default master configuration file, 000000000000.cfg, is shown below:

Example:

```
<?xml version="1.0" standalone="yes"?>
<!-- Default Master SIP Configuration File -->
<!-- Edit and rename this file to <Ethernet-address>.cfg for each
  phone.-->
<!-- $Revision:  1.13  $ $Date:  2004/11/26 23:30:44  $ -->
<APPLICATION APP_FILE_PATH="sip.ld"
    CONFIG_FILES="phone1.cfg, sip.cfg" MISC_FILES=""
    LOG_FILE_DIRECTORY=""/>
```

Master configuration files contain four XML attributes:

APP_FILE_PATH	The path name of the application executable. Has a maximum length of 255 characters. This can be a URL with its own protocol, user name and password, for example http://usr:pwd@server/dir/sip.ld.
CONFIG_FILES	A comma-separated list of configuration files. Each file name has a maximum length of 255 characters and the list of file names has a maximum length of 2047 characters, including commas and white space. Each configuration file can be specified as a URL with its own protocol, user name and password, for example ftp://usr:pwd@server/dir/phone2034.cfg.
MISC_FILES	A comma-separated list of other required files.[a]
LOG_FILE_DIRECTORY	An alternative directory to use for log files if required. This is left blank by default.

a. MISC_FILES is not normally used.

Note

The order of the configuration files listed in CONFIG_FILES is significant.
- The files are processed in the order listed (left to right).
- The same parameters may be included in more than one file.
- The parameter found first in the list of files will be the one that is effective.

This provides a convenient means of overriding the behavior of one or more phones without altering the baseline configuration files for an entire system.

2.2.2.1.1.2 Application Configuration Files

Typically, the files are arranged in the following manner although parameters may be moved around within the files and the file names themselves can be changed as needed.

Per-phone settings ⇨ phoneXXXX.cfg

Application settings ⇨ sip.cfg

Category	Description	Example
Application	Contains parameters that affect the basic operation of the phone such as voice codecs, gains, and tones and the IP address of an application server. All phones in an installation usually share this category of files. This file would normally be modified from Polycom templates.	sip.cfg
User / per-phone	Contains parameters unique to a particular phone user. Typical parameters include: • display name • unique addresses Each phone in an installation usually has its own customized version of user files derived from Polycom templates.	phone1.cfg

These application configuration files dictate the behavior of the phone once it is running the executable specified in the master configuration file.

Important

Configuration files should only be modified by a knowledgeable System Administrator. Applying incorrect parameters may render the phone unusable.

2.2.2.1.2 Deploying a Boot Server for the Phones

The following table describes the steps required for successful deployment of a boot server for SoundPoint® IP and SoundStation® IP phones.

Step:	Instructions:
1. Set up boot server: *Note:* *Typically all phones are configured with the same server account, but the server account provides a means of conveniently partitioning the configuration. Give each account an unique home directory on the server and change the configuration on an account-by-account basis.*	Install boot server application or locate suitable existing server. Use RFC-compliant servers. Create account and home directory.[a] Note that each phone may open multiple connections to the server. The phone will attempt to upload log files, a configuration override file, and a directory file to the server. This requires that the phone's account has delete, write, and read permissions. The phone will still function without these permissions but will not be able to upload files. The files downloaded from the server by the phone should be made read-only.
2. Copy all files:	Copy all files from the distribution zip file to the phone home directory. Maintain the same folder hierarchy.
3. Create per-phone configuration files[b]:	Obtain a list of phone Ethernet addresses (barcoded label on underside of phone). Create per-phone *phoneXXXX.cfg* and *<Ethernet address>.cfg* files by using the 00000000000.cfg and phone1.cfg files from the distribution as templates. Edit contents of *phoneXXXX.cfg* as appropriate. For example, edit the registration parameters. Edit the CONFIG_FILES attribute of the *<Ethernet address>*.cfg files so that it references the appropriate *phoneXXXX.cfg* file. (Replace the reference to phone1.cfg with phoneXXXX.cfg.)

<table>
<tr><th>Step:</th><th>Instructions:</th></tr>
<tr><td>4. Edit sip.cfg:</td><td>See 4.6 Configuration Files on page 70, particularly for SIP server address.

Most of the default settings are typically adequate, however, if overriding SNTP settings are not available via DHCP, the SNTP GMT offset and (possibly) the SNTP server address will need to be edited for the correct local conditions. Changing the default daylight savings parameters will likely be necessary outside of North American locations.

(Optional) Disable the local web (HTTP) server or alter its signalling port if local security policy dictates.

Change the default location settings:
• user interface language
• time and date format</td></tr>
<tr><td>5. Decide on boot server security policy:</td><td>Polycom recommends allowing file uploads to the boot server where the security environment permits. This allows event log files to be uploaded and changes made by the phone user to the configuration (via the web server and local user interface) and changes made to the directory to be backed up.

For organizational purposes, configuring a separate log file directory is recommended, but not required (see LOG_FILE_DIRECTORY in 2.2.2.1.1.1 Master Configuration Files on page 12).

File permissions should give the minimum access required, and the account used should have no other rights on the server.

The phone's server account needs to be able to add files to which it can write in the log file directory and the root directory. It must also be able to list files in all directories mentioned in the [mac].cfg file. All other files that the phone needs to read, such as the application executable and the standard configuration files, should be made read-only via file server file permissions.</td></tr>
</table>

Step:	Instructions:
6. Reboot phones after configuring their boot server via DHCP or statically:	See 2.2.1 Basic Network Setup on page 4. To reboot phones, a menu option can be selected or a key combination can be held down. The menu option is called Restart Phone and it is in the Settings menu. For the key combination, press and hold the following keys simultaneously until a confirmation tone is heard or for about three seconds: IP 300 & IP 301: Volume-, Volume+, Hold and Do Not Disturb IP 500 & IP 501: Volume-, Volume+, Hold, and Messages IP 600 & IP 601: Volume-, Volume+, Mute, and Messages IP 4000: *, #, Volume+, and Select Monitor the boot server event log and the uploaded event log files (if permitted): Ensure that the configuration process completed correctly. Start making calls!

a. If the provisioning protocol requires an account name and password, the server account name and password must match those configured in the phones. Defaults are: provisioning protocol: FTP, name: PlcmSpIp, password: PlcmSpIp

b. This step may be omitted if per-phone configuration is not needed.

2.2.2.2 Local Phone Configuration

As the only method of modifying phone configuration or as a distributed method of augmenting a centralized provisioning model, a local phone-based configuration web server is available, unless disabled via sip.cfg. For more information, see 4.6.1.11 Web Server <HTTPD/> on page 107. The phone's local user interface also permits many application settings to be modified, such as SIP server address or ring type or regional settings such as time/date format and language.

Local Web Server Access	Point your web browser to http://<phoneIPAddress>/. Configuration pages are accessible from the menu along the top banner. The web server will issue an authentication challenge to all pages except for the home page. Credentials are (case sensitive): • User Name: Polycom • Password: The administrator password is used for this.
Local Settings Menu Access	Some items in the Settings menu are locked to prevent accidental changes. To unlock these menus, enter the user or administrator passwords. The administrator password can be used anywhere that the user password is used. Factory default passwords are: • User password: 123 • Administrator password: 456
Passwords:	
Administrator password required.	Network Configuration SIP Configuration SSL Security settings Reset to Default - local configuration, device settings, and file system format
User password required.	Restart Phone

Changes made via the web server or local user interface are stored internally as overrides. These overrides take precedence over settings contained in the configuration obtained from the boot server that existed previously within the phone.

If the boot server permits uploads, these override setting will be saved in a file called <*Ethernet address*>-phone.cfg on the boot server.

Important

Local configuration changes will continue to override the boot server-derived configuration until deleted via the Reset User Settings menu selection.

3 Features

This section describes the many features and corresponding administration points of SoundPoint® IP and SoundStation® IP. References are made frequently to 4.6 Configuration Files on page 71.

3.1 Basic Features

3.1.1 Call Log

The phone maintains a call log. The log:
- contains call information such as remote party identification, time and date, and call duration,
- allows for convenient redialing of previous outgoing calls and for returning incoming calls,
- can be used to save contact information from call log entries to the contact directory.

The call log is stored in volatile memory and is maintained automatically by the phone in three separate lists: Missed Calls, Received Calls and Placed Calls. The call lists can be cleared manually by the user and will be erased on reboot.

Central (boot server)	Configuration File: sip.cfg	Enable or disable all call lists or individual call lists. • For more information, see 4.6.1.23 Feature <feature/> on page 125.
Local	Web Server (if enabled)	None.
	Local Telephone User Interface	None.

3.1.2 Call Timer

A call timer is provided on the display. A separate call timer is maintained for each distinct call in progress.

3.1.3 Call Waiting

When an incoming call arrives while the user is active on another call, the incoming call is presented to the user visually on the LCD display. A configurable sound effect such as the familiar call-waiting beep will be mixed with the active call audio as well.

3.1.4 Called Party Identification

The phone displays and logs the identity of the remote party specified for outgoing calls. This is the party that the user intends to connect with.

3.1.5 Calling Party Identification

The phone displays the caller identity, derived from the network signalling, when an incoming call is presented. For calls from parties for which a directory entry exists, the local name assigned to the directory entry may optionally be substituted.

Central (boot server)	Configuration File: sip.cfg	Specify whether or not to use directory name substitution. • For more information, see 4.6.1.4 User Preferences <user_preferences/> on page 82.
Local	Web Server (if enabled)	Specify whether or not to use directory name substitution. Navigate to: http://<phoneIPAddress>/coreConf.htm#us Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. Changes will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Telephone User Interface	None.

3.1.6 Missed Call Notification

The phone can display the number of calls missed since the user last looked at the Missed Calls list. The types of calls which are counted as “missed” can be configured per registration. Remote missed-call notification can be used to notify the phone when a call originally destined for it is diverted by another entity such as a SIP server.

Central (boot server)	Configuration file: sip.cfg	Turn this feature on or off. • For more information, see 4.6.1.23 Feature <feature/> on page 125.
	Configuration file: phone1.cfg	Specify per-registration whether all missed-call events or only remote/server-generated missed-call events will be displayed. • For more information, see 4.6.2.2.3 Missed Call Configuration <serverMissedCall/> on page 132.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

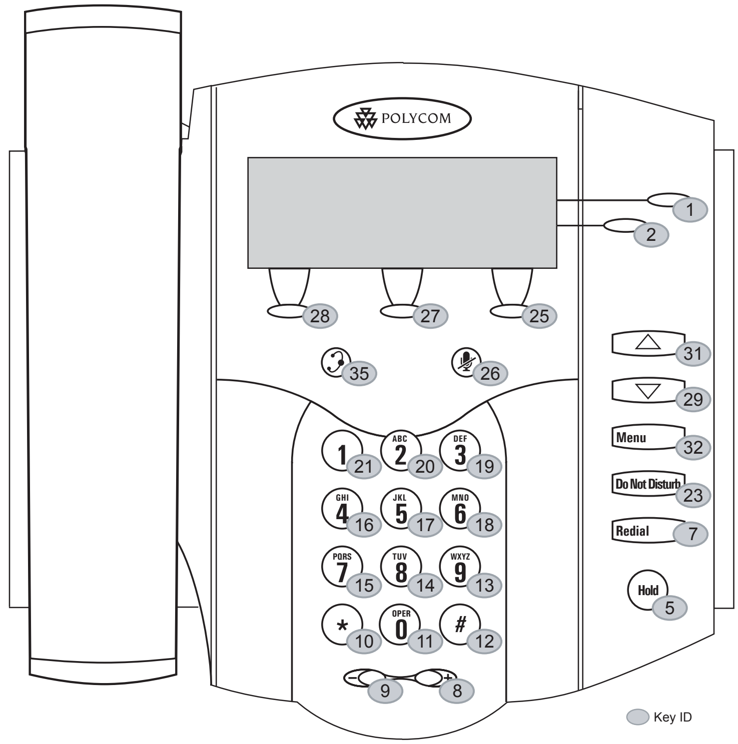
3.1.7 Configurable Feature Keys

All key functions can be changed from the factory defaults, although this is typically not necessary. The scrolling timeout for specific keys can be configured.

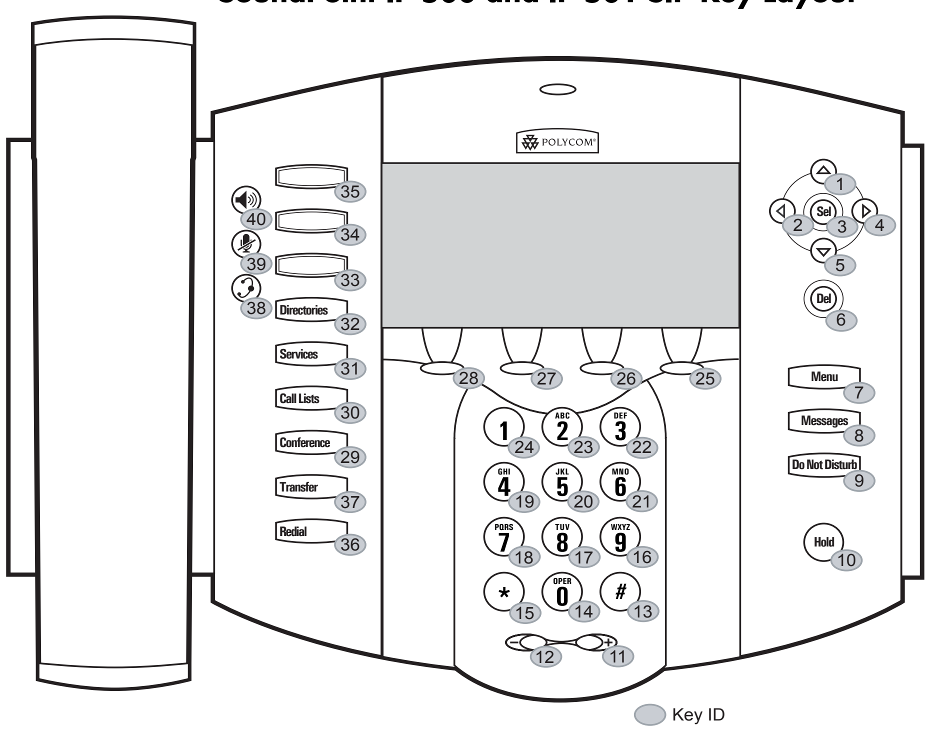
Central (boot server)	Configuration File: sip.cfg	Set the key scrolling timeout, key functions, and sub-pointers for each key (usually not necessary). • For more information, see 4.6.1.15 Keys <keys/> on page 113.
Local	Web Server (if enabled)	None.
	Local Telephone User Interface	None.

The following diagrams and table show the default SIP key layouts for SoundPoint® IP 300, IP 301, IP 500, IP 501, IP 600, IP 601 and SoundStation® IP 4000 models.

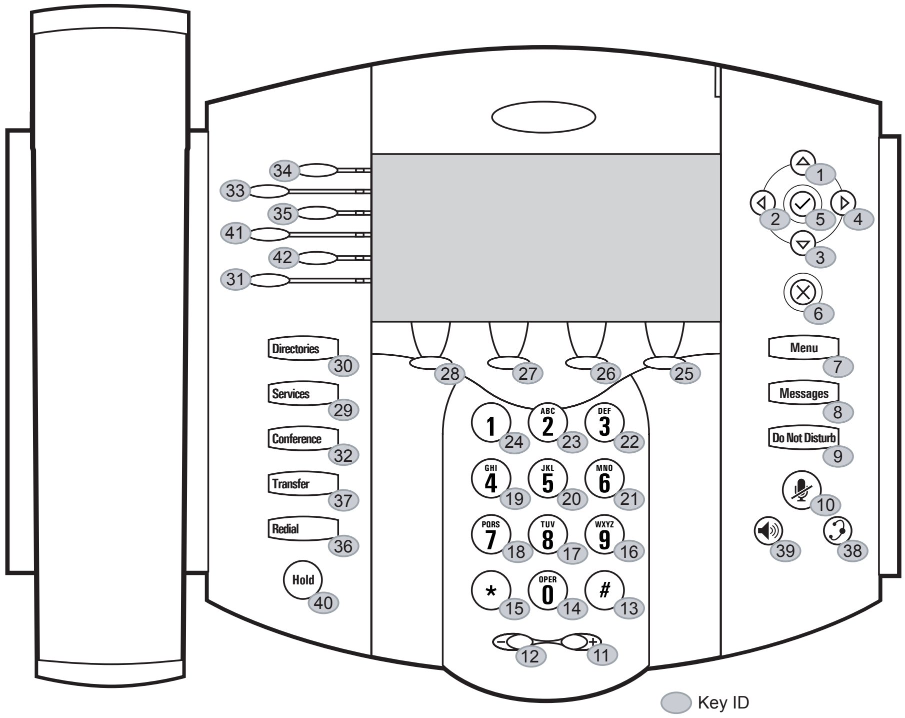
SoundPoint IP 300 and IP 301 SIP Key Layout



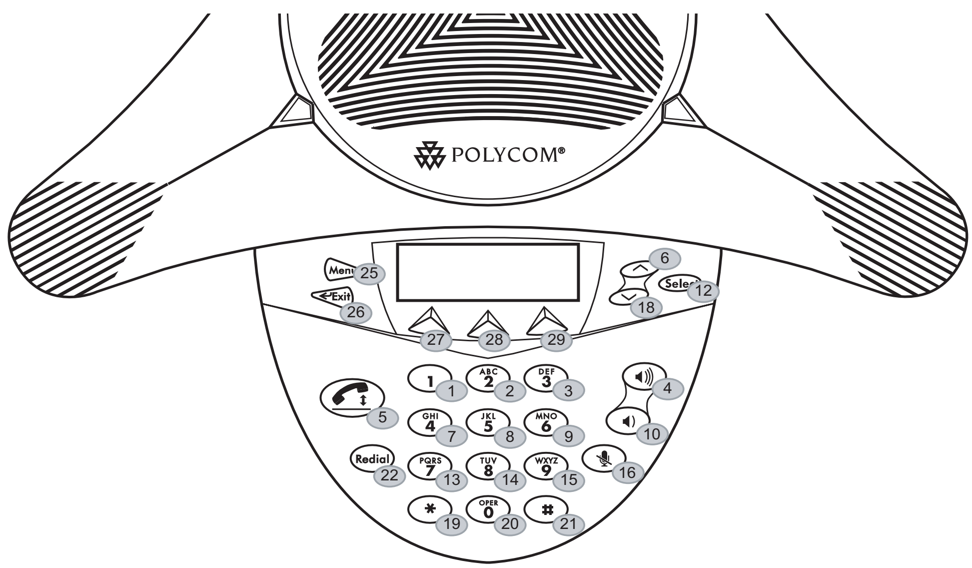
SoundPoint IP 500 and IP 501 SIP Key Layout



SoundPoint IP 600 and IP 601 SIP Key Layout



SoundStation IP 4000 SIP Key Layout



Key ID	IP 300 & 301 Function	IP 500 & 501 Function	IP 600 & 601 Function	IP 4000 Function
1	Line1	ArrowUp	ArrowUp	Dialpad1
2	Line2	ArrowLeft	ArrowLeft	Dialpad2
3	n/a	Select	ArrowDown	Dialpad3
4	n/a	ArrowRight	ArrowRight	VolUp
5	Hold	ArrowDown	Select	Handsfree
6	n/a	Delete	Delete	ArrowUp
7	DoNotDisturb	Menu	Menu	Dialpad4
8	VolUp	Messages	Messages	Dialpad5
9	VolDown	DoNotDisturb	DoNotDisturb	Dialpad6
10	DialpadStar	Hold	MicMute	VolDown
11	Dialpad0	VolUp	VolUp	n/a
12	DialpadPound	VolDown	VolDown	Select
13	Dialpad9	DialpadPound	DialpadPound	Dialpad7
14	Dialpad8	Dialpad0	Dialpad0	Dialpad8
15	Dialpad7	DialpadStar	DialpadStar	Dialpad9
16	Dialpad4	Dialpad9	Dialpad9	MicMute
17	Dialpad5	Dialpad8	Dialpad8	n/a
18	Dialpad6	Dialpad7	Dialpad7	ArrowDown
19	Dialpad3	Dialpad4	Dialpad4	DialpadStar
20	Dialpad2	Dialpad5	Dialpad5	Dialpad0
21	Dialpad1	Dialpad6	Dialpad6	DialpadPound
22	n/a	Dialpad3	Dialpad3	Redial
23	Redial	Dialpad2	Dialpad2	n/a
24	n/a	Dialpad1	Dialpad1	n/a
25	SoftKey3	SoftKey4	SoftKey4	Menu
26	MicMute	SoftKey3	SoftKey3	Exit
27	SoftKey2	SoftKey2	SoftKey2	SoftKey1
28	SoftKey1	SoftKey1	SoftKey1	SoftKey2
29	ArrowDown	Conference	Services	SoftKey3
30	n/a	CallHistory	Directories	n/a

Key ID	IP 300 & 301 Function	IP 500 & 501 Function	IP 600 & 601 Function	IP 4000 Function
31	ArrowUp	Services	Line6	n/a
32	Menu	Directories	Conference	n/a
33	n/a	Line3	Line2	n/a
34	n/a	Line2	Line1	n/a
35	Headset	Line1	Line3	n/a
36	n/a	Redial	Redial	n/a
37	n/a	Transfer	Transfer	n/a
38	n/a	Headset	Headset	n/a
39	n/a	MicMute	Handsfree	n/a
40	n/a	Handsfree	Hold	n/a
41	n/a	n/a	Line4	n/a
42	n/a	n/a	Line5	n/a

3.1.8 Connected Party Identification

Where possible, the identity of the remote party to which the user has connected is displayed and logged. The connected party identity is derived from the network signaling. In some cases the remote party will be different from the called party identity due to network call diversion.

3.1.9 Context Sensitive Volume Control

The volume of user interface sound effects, such as the ringer, and the receive volume of call audio is adjustable. While transmit levels are fixed according to the TIA/EIA-810-A standard, receive volume is adjustable. For SoundPoint® IP, if using the default configuration parameters, the receive handset/headset volume resets to nominal after each call to comply with regulatory requirements. See 4.6.1.8.2 Volume Persistence <volume/> on page 94.

3.1.10 Customizable Audio Sound Effects

Audio sound effects used for incoming call alerting and other indications are customizable. Sound effects can be composed of patterns of synthesized tones or sample audio files. The default sample audio files may be replaced with alternates in .wav file format. Supported .wav formats include:

- mono G.711 (13-bit dynamic range, 8-khz sample rate),
- mono L16/16000[2] (16-bit dynamic range, 16-kHz sample rate).

Note

The alternate sampled audio sound effect files must be present on the boot server or the Internet for downloading at boot time.

Central (boot server)	Configuration File: sip.cfg	Specify patterns used for sound effects and the individual tones or sampled audio files used within them. For more information, see: • 4.6.1.3.3 Call Progress Tones <callProgTones> on page 81, • 4.6.1.6 Sampled Audio for Sound Effects <sampled_audio/> on page 85, • 4.6.1.7 Sound Effects <sound_effects/> on page 87.
Local	Web Server (if enabled)	Specify sampled audio wave files to replace the built-in defaults. Navigate to: http://<phoneIPAddress>/coreConf.htm#sa Changes are saved to local flash and backed up to <*Ethernet address*>phone-.cfg on the boot server and will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	None.

3.1.11 Message Waiting Indication

The phone will flash a message-waiting indicator LED when instant messages are waiting, and it can be configured to do so when voice messages are waiting.

3.1.12 Distinctive Incoming Call Treatment

The phone can automatically apply distinctive treatment to calls containing specific attributes. The distinctive treatment that can be applied includes customizable alerting sound effects and automatic call diversion or rejection. Call attributes that can trigger

2. L16/16000 is not supported on SoundPoint® IP 300, 301 and SoundStation® IP 4000 phones.

distinctive treatment include the calling party name or SIP contact (number or URL format).

Administration: Distinctive Incoming Call Treatment

For more information, see 3.1.17 Local Contact Directory on page 29.

3.1.13 Distinctive Ringing

There are three aspects to Distinctive Ringing:

1. The user can select the ring type for each line. There are many different ring patterns to choose from.
2. The ring type for specific callers can be assigned in the contact directory. For more information, see 3.1.12 Distinctive Incoming Call Treatment on page 26. This feature has higher priority than Item 1.
3. The SIP Alert-Info field can be used to map calls to specific ring types. This feature has higher priority than Items 1 and 2.

Central (boot server)	Configuration file: sip.cfg	Specify the mapping of Alert-Info strings to ring types. • For more information, see 4.6.1.1.3.2 Alert Information <alertInfo/> on page 74.
	Configuration file: phone1.cfg	Specify the ring type to be used for each line. • For more information, see 4.6.2.1 Registration <reg/> on page 128.
	XML File: <Ethernet address>-directory.xml	This file can be created manually using an XML editor. • For more information, see 3.1.17.1 Local Contact Directory File Format on page 31.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	The user can edit the ring types selected for each line under the Settings menu. The user can also edit the directory contents. Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. These changes will permanently override global settings unless deleted via the Reset User Settings menu selection.

3.1.14 Distinctive Call Waiting

The SIP Alert-Info field can be used to map calls to distinct call waiting types, currently limited to two styles.

Central (boot server)	Configuration file: sip.cfg	Specify the mapping of Alert-Info strings to call waiting types. • For more information, see 4.6.1.1.3.2 Alert Information <alertInfo/> on page 74.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.1.15 Do-Not-Disturb

A do-not-disturb feature is available to temporarily stop all incoming call alerting. Calls can optionally be treated as though the phone is busy while Do-Not-Disturb (DND) is enabled. Incoming calls received while DND is enabled are logged as missed.

Central (boot server)	Configuration file: sip.cfg	Specify whether or not DND results in incoming calls being given busy treatment. • For more information, see 4.6.1.12 Call Handling Configuration <call/> on page 108.
	Configuration file: phone1.cfg	Specify whether DND is treated as a per-registration feature or a global feature on the phone. • For more information, see 4.6.2.2.1 Do Not Disturb <donotdisturb/> on page 131.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	Enable or disable DND using the “Do Not Disturb” key on the SoundPoint IP 300, 301, 500, 501 and 600 or the Features menu on the SoundStation IP 4000.

3.1.16 Handset, Headset, and Speakerphone

SoundPoint® IP phones come standard with a handset and a dedicated connector is provided for a headset (not supplied). The SoundPoint® IP 500, 501, 600 and 601 phones have full-duplex speakerphones. The SoundPoint® IP 300 and 301 phones have a listen-only speakerphone. The SoundPoint® phones provide dedicated keys for convenient selection of either the speakerphone or headset. The SoundStation® IP 4000 phones are full-duplex speakerphones.

Central (boot server)	Configuration file: sip.cfg	Enable or disable persistent headset mode. • For more information, see 4.6.1.4 User Preferences <user_preferences/> on page 82.
Local	Web Server (if enabled)	Enable or disable persistent headset mode. Navigate to: http://<phoneIPAddress>/coreConf.htm#us
	Local Phone User Interface	Enable or disable persistent headset mode via the Settings menu. Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. Changes will permanently override global settings unless deleted via the Reset User Settings menu.

3.1.17 Local Contact Directory

The phone maintains a local contact directory. The directory can be downloaded from the boot server and edited locally. Contact information from previous calls may be easily added to the directory for convenient future access. The directory is the central database for several other features including speed-dial, distinctive incoming call treatment, presence, and instant messaging.

See the following table for further information.

Central (boot server)	Configuration file: sip.cfg	Set whether the directory uses volatile storage on the phone (required on the IP 500 platform for directories greater than 25 entries). • For more information, see 4.6.1.13 Directory <directory/> on page 110.
	XML file: 000000000000-directory.xml	A sample file named 000000000000-directory~.xml (Note extra "~" in the file name) is included with the application file distribution. This file can be used as a template for the per-phone <*Ethernet address*>-directory.xml directories (edit contents then rename to <*Ethernet address*>-directory.xml). It also can be used to seed new phones with an initial directory (edit contents than remove "~" from file name). Telephones without a local directory, such as new units from the factory, will download the 00000000000-directory.xml directory and base their initial directory on it. These files should be edited with an XML editor. • For information on file format, see 3.1.17.1 Local Contact Directory File Format on page 31.
	XML file: <Ethernet address>-directory.xml	This file can be created manually using an XML editor. • For information on file format, see 3.1.17.1 Local Contact Directory File Format on page 31.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	The user can edit the directory contents at will. Changes will be stored in the phone's flash file system and backed up to the boot server copy of <*Ethernet address*>-directory.xml if this is configured. When the phone boots, the boot server copy of the directory, if present, will overwrite the local copy.

3.1.17.1 Local Contact Directory File Format

An example local contact directory is shown. Look to the table for an explanation of each element.

Local Contact Directory File example:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<directory>
  <item_list>
    <item>
      <ln>Doe</ln>
      <fn>John</fn>
      <ct>1001</ct>
      <sd>1</sd>
      <rt>1</rt>
      <dc />
      <ad>0</ad>
      <ar>0</ar>
      <bw>0</bw>
      <bb>0</bb>
    </item>
      • • •
    <item>
      <ln>Smith</ln>
      <fn>Bill</fn>
      <ct>1003</ct>
      <sd>3</sd>
      <rt>3</rt>
      <dc />
      <ad>0</ad>
      <ar>0</ar>
      <bw>0</bw>
      <bb>0</bb>
    </item>
  </item_list>
</directory>
```

Element	Permitted Values	Interpretation
fn	UTF-8 encoded string of up to 40 bytes[a]	first name
ln	UTF-8 encoded string of up to 40 bytes	last name

Element	Permitted Values	Interpretation
ct	UTF-8 encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL	contact *Cannot be Null or duplicated*; is used by the phone to address a remote party in the same way that a string of digits or a SIP URL are dialed manually by the user. This element is also used to associate incoming callers with a particular directory entry.
sd	Null, 1 to 40	speed-dial index Associates a particular entry with a speed dial bin for one-touch dialing or dialing from the speed dial menu.
rt	Null, 1 to 21	ring type When incoming calls can be associated with a directory entry by matching the address fields, this field is used to specify ring type to be used.
dc	UTF-8 encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL	divert contact The forward-to address for the autodivert feature.
ad	0,1	auto divert If 1, automatically diverts callers that match the directory entry to the address specified in divert-contact.
ar	0,1	auto-reject[b] If 1, automatically rejects callers that match the directory entry.
bw	0,1	buddywatching If 1, add this contact to the list of watched phones.
bb	0,1	buddyblock If 1, block this contact from watching this phone.

a. In some cases, this will be less than 40 characters due to UTF-8's variable length encoding.
b. If auto-divert is also enabled, it has precedence over auto-reject.

3.1.18 Local Digit Map

The phone has a local digit map feature to automate the setup phase of number-only calls. When properly configured, this feature eliminates the need for using the **Send** soft key when making outgoing calls. Instead, as soon as a digit pattern matching the digit map is found, the call setup process will complete automatically. This feature is similar to the digit map feature of the Media Gateway Control Protocol (MGCP) and the configuration syntax is the same as that specified in 2.1.5 of RFC 3435. The phone behavior when the user dials digits that do not match the digit map is configurable. It is also possible to strip a trailing # from the digits sent.

Central (boot server)	Configuration file: sip.cfg	Specify impossible match behavior, trailing # behavior, digit map matching strings, and time out value. • For more information, see 4.6.1.2 Dial Plan <dial-plan/> on page 77.
	Configuration file: phone1.cfg	Specify per-registration impossible match behavior, trailing # behavior, digit map matching strings, and time out values that override those in sip.cfg. • For more information, see 4.6.2.4 Dial Plan <dial-plan/> on page 135.
Local	Web Server (if enabled)	Specify impossible match behavior, trailing # behavior, digit map matching strings, and time out value. Navigate to: http://<phoneIPAddress>/appConf.htm#ls Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. Changes will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	None.

3.1.19 Microphone Mute

A microphone mute feature is provided. When activated, visual feedback is provided. This is a local function and cannot be overridden by the network.

3.1.20 Multiple Line Keys per Registration

More than one line key can be allocated to a single registration (phone number or line). The number of line keys allocated per registration is configurable.

Central (boot server)	Configuration file: phone1.cfg	Specify the number of line keys to assign per registration. • For more information, see 4.6.2.1 Registration <reg/> on page 128.

Local	Web Server (if enabled)	Specify the number of line keys to assign per registration. Navigate to: http://<phoneIPAddress>/reg.htm Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. They will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	Specify the number of line keys to assign per registration using the SIP Configuration menu. Either the Web Server or the boot server configuration files or the local phone user interface should be used to configure registrations, not a mixture of these options. When the SIP Configuration menu is used, it is assumed that all registrations use the same server.

3.1.21 Multiple Call Appearances

The phone supports multiple concurrent calls. The hold feature can be used to pause activity on one call and switch to another call. The number of concurrent calls per line key is configurable. Each registration can have more than one line key assigned to it, see 3.1.20 Multiple Line Keys per Registration on page 33.

Central (boot server)	Configuration file: sip.cfg	Specify the default number of calls which can be active or on hold per line key. • For more information, see 4.6.1.12 Call Handling Configuration <call/> on page 108.
	Configuration file: phone1.cfg	Specify per-registration the number of calls which can be active or on hold per line key assigned to that registration. This will override the default value specified in sip.cfg. • For more information, see 4.6.2.1 Registration <reg/> on page 128.

Local	Web Server (if enabled)	Specify the default number of calls which can be active or on hold per line key and the number of calls per registration which can be active or on hold per line key assigned to that registration. Navigate to: http://<phoneIPAddress>/appConf.htm#ls and http://<phoneIPAddress>/reg.htm Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. They will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	Specify per-registration the number of calls which can be active or on hold per line key assigned to that registration using the SIP Configuration menu. Either the Web Server or the boot server configuration files or the local phone user interface should be used to configure registrations, not a mixture of these options. When the SIP Configuration menu is used, it is assumed that all registrations use the same server.

3.1.22 Shared Call Appearances

Calls and lines on multiple phones can be logically related to each other. A call that is active on one phone will be presented visually to phones which share that call appearance. Mutual exclusion features emulate traditional PBX or key system privacy for shared calls. Incoming calls can be presented to multiple phones simultaneously. This feature is dependent on support from a SIP server which binds the appearances together logically and looks after the necessary state notifications and performs an access control function. For more information, see 5.2.4 Shared Call Appearance Signaling on page 149.

See the following table for further information.

Central (boot server)	Configuration file: sip.cfg	Specify whether diversion should be disabled on shared lines. • For more information, see 4.6.1.12.1 Shared Calls <shared/> on page 109. Specify line-seize subscription period. • For more information, see 4.6.1.1.2 Server <server/> on page 71. Specify standard or non-standard behavior for processing line-seize subscription for mutual exclusion feature. • For more information, see 4.6.1.1.3.4 Special Events <specialEvent/> on page 76.
	Configuration file: phone1.cfg	Specify per-registration line type (private or shared) and line-seize subscription period if using per-registration servers. A shared line will subscribe to a server providing call state information. • For more information, see 4.6.2.1 Registration <reg/> on page 128. Specify per-registration whether diversion should be disabled on shared lines. • For more information, see 4.6.2.3 Diversion <divert/> on page 133.

Local	Web Server (if enabled)	Specify line-seize subscription period. Navigate to: http://<phoneIPAddress>/appConf.htm#se Specify standard or non-standard behavior for processing line-seize subscription for mutual exclusion feature. Navigate to: http://<phoneIPAddress>/appConf.htm#ls Specify per-registration line type (private or shared) and line-seize subscription period if using per-registration servers, and whether diversion should be disabled on shared lines. Navigate to: http://<phoneIPAddress>/reg.htm Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. They will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	Specify per-registration line type (private or shared) using the SIP Configuration menu. Either the Web Server or the boot server configuration files or the local phone user interface should be used to configure registrations, not a mixture of these options. When the SIP Configuration menu is used, it is assumed that all registrations use the same server.

3.1.23 Bridged Line Appearances

Calls and lines on multiple phones can be logically related to each other. A call that is active on one phone will be presented visually to phones which share that line. Mutual exclusion features emulate traditional PBX or key system privacy for shared calls. Incoming calls can be presented to multiple phones simultaneously. This feature is dependent on support from a SIP server which binds the appearances together logically and looks after the necessary state notifications and performs an access control function. For more information, see 5.2.5 Bridged Line Appearance Signaling on page 149.

See the following table for further information.

Note: In the configuration files, bridged lines are configured by “shared line” parameters.

Central (boot server)	Configuration file: sip.cfg	Specify whether diversion should be disabled on shared lines. • For more information, see 4.6.1.12 Call Handling Configuration <call/> on page 108.
	Configuration file: phone1.cfg	Specify per-registration line type (private or shared) and the shared line third party name. A shared line will subscribe to a server providing call state information. • For more information, see 4.6.2.1 Registration <reg/> on page 128. Specify per-registration whether diversion should be disabled on shared lines. • For more information, see 4.6.2.3 Diversion <divert/> on page 133.
Local	Web Server (if enabled)	Specify per-registration line type (private or shared) and third party name, and whether diversion should be disabled on shared lines. Navigate to: http://<phoneIPAddress>/reg.htm Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. They will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	Specify per-registration line type (private or shared) and the shared line third party name using the SIP Configuration menu. Either the Web Server or the boot server configuration files or the local phone user interface should be used to configure registrations, not a mixture of these options. When the SIP Configuration menu is used, it is assumed that all registrations use the same server.

3.1.24 Customizable Fonts and Indicators

The phone’s user interface can be customized by changing the fonts and graphic icons used on the display and the LED indicator patterns. Pre-existing fonts embedded in the software can be overwritten or new fonts can be downloaded. The bitmaps and bitmap animations used for graphic icons on the display can be changed and repositioned. LED flashing sequences and colors can be changed.

See the following table for further information.

Central (boot server)	Configuration File: sip.cfg	Specify fonts to overwrite existing ones or specify new fonts. • For more information, see 4.6.1.14 Fonts <font/> on page 111. Specify which bitmaps to use. • For more information, see 4.6.1.16 Bitmaps <bitmaps/> on page 115. Specify how to create animations and LED indicator patterns. • For more information, see 4.6.1.17 Indicators <indicators/> on page 116.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.1.25 Soft Key-Driven User Interface

The user interface makes extensive use of intuitive, context-sensitive soft key menus.

3.1.26 Speed Dial

Entries in the local directory can be linked to the speed dial system. The speed dial system allows calls to be placed quickly from dedicated keys as well as from a speed dial menu. If Presence watching is enabled for speed dial entries, their status will be shown on the idle display if the SIP server supports this feature. See 3.4.1 Presence on page 51.

See the following table for further information.

Central (boot server)	XML file: <Ethernet address>-directory.xml	The <sd>**x**</sd> element in the <*Ethernet address*>-directory.xml file links a directory entry to a speed dial resource within the phone. Speed dial entries are mapped automatically to unused line keys (line keys are not available on the IP 4000) and are available for selection within the speed dial menu. (Press the up-arrow key from the idle display to jump to SpeedDial). • For more information, see 3.1.17.1 Local Contact Directory File Format on page 31.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	The next available Speed Dial Index is assigned to new directory entries. Key-pad short cuts are available to facilitate assigning and modifying the Speed Dial Index value for entries in the directory. The Speed Dial Index field is used to link directory entries to speed dial operations. Changes will be stored in the phone's flash file system and backed up to the boot server copy of <*Ethernet address*>-directory.xml if this is configured. When the phone boots, the boot server copy of the directory, if present, will overwrite the local copy.

3.1.27 Time and Date Display

The phone maintains a local clock and calendar. Time and date can be displayed in certain operating modes such as when the phone is idle and during a call. The clock and calendar must be synchronized to a remote SNTP timeserver. The time and date displayed on the phone will flash continuously until a successful SNTP response is received to indicate that they are not accurate. The time and date display can use one of several different formats and can be turned off.

Central (boot server)	Configuration file: sip.cfg	Turn time and date display on or off. • For more information, see 4.6.1.4 User Preferences <user_preferences/> on page 82. Set the time and date display formats. • For more information, see 4.6.1.3.2 Date and Time <datetime/> on page 81. Set the basic SNTP settings and daylight savings parameters. • For more information, see 4.6.1.10.2 Time Synchronization <SNTP/> on page 104.
Local	Web Server (if enabled)	Set the basic SNTP and daylight savings settings. Navigate to: http://<phoneIPAddress>/coreConf.htm#ti Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. They will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	The basic SNTP settings can be made in the Network Configuration menu. • For more information, see 2.2.1.1 DHCP or Manual TCP/IP Setup on page 5. The user can edit the time and date format and enable or disable the time and date display under the Settings menu. Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. They will permanently override global settings unless deleted via the Reset User Settings menu selection.

3.1.28 Idle Display Animation

All phones except the SoundPoint® IP 300 and SoundPoint® IP 301 can display a customized animation on the idle display in addition to the time and date. For example, a company logo could be displayed.

Central (boot server)	Configuration file: sip.cfg	To turn idle display animation on or off. • For more information, see 4.6.1.17 Indicators <indicators/> on page 116. To replace the animation used for the idle display. • For more information, see 4.6.1.17.1 Animations <Animations/> <IP_300/>, <IP_500/>, <IP_600/> and <IP_4000/> on page 116. To change the position of the idle display animation. • For more information, see 4.6.1.17.4.2 Graphic Icons <gi/> <IP_300/>, <IP_500/>, <IP_600/> and <IP_4000/> on page 118.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.2 Call Management Features

3.2.1 Automatic Off-hook Call Placement

The phone supports an optional automatic off-hook call placement feature for each registration.

Central (boot server)	Configuration file: phone1.cfg	Specify which registrations have the feature and what contact to call when going off hook. • For more information, see 4.6.2.2.2 Automatic Off-hook Call Placement <autoOffHook/> on page 131.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.2.2 Call Hold

Call hold is a fundamental feature of the phone. The purpose of hold is to pause activity on one call so that the user may use the phone for another task, such as to make or receive another call. Network signaling is employed to request that the remote party stop sending media and to inform them that they are being held. A configurable local hold reminder feature can be used to remind the user that they have placed calls on hold.

Central (boot server)	Configuration file: sip.cfg	Specify whether RFC 2543 (c=0.0.0.0) or RFC 3264 (a=sendonly or a=inactive) outgoing hold signaling is used. • For more information, see 4.6.1.1.3 SIP <SIP/> on page 73. Specify local hold reminder options. • For more information, see 4.6.1.12.2 Hold, Local Reminder <hold/><localReminder/> on page 109.
Local	Web Server (if enabled)	Specify whether or not to use RFC 2543 (c=0.0.0.0) outgoing hold signaling. The alternative is RFC 3264 (a=sendonly or a=inactive). Navigate to: http://<phoneIPAddress>/appConf.htm#ls Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. They will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	Use the SIP Configuration menu to specify whether or not to use RFC 2543 (c=0.0.0.0) outgoing hold signaling. The alternative is RFC 3264 (a=sendonly or a=inactive).

3.2.3 Call Transfer

Call transfer enables the user (User A or transferring user) to transform an existing call with User B (primary call) into a new call between User B and a third user C (transferred-to user) selected by User A. The phone offers three types of transfers;

- Blind transfers: The call is transferred immediately to C after A has finished dialing C's number. User A does not hear ring-back.
- Consultation transfers which are dispatched during the proceeding state: User A dials C's number and hears ring-back and decides to complete the transfer before C answers. This option can be disabled.
- True consultation transfers: User A dials C's number and consults privately with C after the call is answered and then completes the transfer.

Central (boot server)	Configuration file: sip.cfg	Specify whether to allow a transfer during the proceeding state of a consultation call. • For more information, see 4.6.1.1.3 SIP <SIP/> on page 73.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.2.4 Three-Way Conference, Local or Centralized

Local or centralized conferences[3] are supported. The phone can conference together the local user with the remote parties of two independent calls by using the phone's local audio processing resources for the audio bridging. For a local conference there is no dependency on network signaling.

The phone also supports centralized conferences for which external resources are used such as a conference bridge. This depends on network signaling.

Central (boot server)	Configuration file: sip.cfg	Specify which type of conference to establish and the address of the centralized conference resource. • For more information, see 4.6.1.1.3.5 Conference Setup <conference/> on page 76.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.2.5 Call Diversion (Call Forward)

The phone provides a flexible call diversion feature to divert (forward) calls to another destination. Call diversion can be applied automatically to all calls, only when the phone is busy, or after an extended period of alerting. The user can elect to manually divert calls while they are in the alerting state to a predefined or manually specified destination. The call diversion feature works in conjunction with the distinctive

3. On SoundStation IP® 4000, conferences are not available if the G.729 codec is enabled on the phone. This restriction will be removed in future releases.

incoming call treatment feature. The user's ability to originate calls is unaffected by all call diversion options. Each registration has its own diversion properties.

Central (boot server)	Configuration file: phone1.cfg	Set all call diversion settings including a global forward-to contact and individual settings for call forward all, call forward busy, call forward no-answer, and call forward do-not-disturb. • For more information, see 4.6.2.3 Diversion <divert/> on page 133.
Local	Web Server (if enabled)	Set all call diversion settings. Navigate to: http://<phoneIPAddress>/reg.htm Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. They will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	The user can set the call-forward-all setting from the idle display (enable/disable and specify the forward-to contact) as well as divert callers while the call is alerting. Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. They will permanently override global settings unless deleted via the Reset User Settings menu selection.

3.2.6 Directed Call Pick-up

Calls to another phone can be picked up by dialing the extension of the other phone. This feature depends on support from a SIP server.

Central (boot server)	Configuration file: sip.cfg	Turn this feature on or off. • For more information, see 4.6.1.23 Feature <feature/> on page 125.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.2.7 Group Call Pick-up

Calls to another phone within a pre-defined group can be picked up without dialing the extension of the other phone. This feature depends on support from a SIP server.

Central (boot server)	Configuration file: sip.cfg	Turn this feature on or off. • For more information, see 4.6.1.23 Feature <feature/> on page 125.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.2.8 Call Park / Retrieve

An active call can be parked, and the parked call can be retrieved by another phone. This feature depends on support from a SIP server.

Central (boot server)	Configuration file: sip.cfg	Turn this feature on or off. • For more information, see 4.6.1.23 Feature <feature/> on page 125.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.2.9 Last Call Return

The phone allows server-based last call return. This feature depends on support from a SIP server.

Central (boot server)	Configuration file: sip.cfg	Turn this feature on or off. • For more information, see 4.6.1.23 Feature <feature/> on page 125. Specify the string sent to the server for last-call-return. • For more information, see 4.6.1.12 Call Handling Configuration <call/> on page 108.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.3 Audio Processing Features

Proprietary state-of-the-art digital signal processing (DSP) technology is used to provide an excellent audio experience.

3.3.1 Low-Delay Audio Packet Transmission

The phone is designed to minimize latency for audio packet transmission.

3.3.2 Jitter Buffer and Packet Error Concealment

The phone employs a high-performance jitter buffer and packet error concealment system designed to mitigate packet inter-arrival jitter and out-of-order or lost (lost or excessively 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.

Central (boot server)	Configuration file: sip.cfg	Set the jitter buffer tuning parameters including minimum and maximum size and shrink aggression. • For more information, see 4.6.1.8.1.2 Codec Profiles <profiles/> on page 93.
Local	Web Server (if enabled)	Set the jitter buffer tuning parameters including minimum and maximum size and shrink aggression. Navigate to: http://<phoneIPAddress>/coreConf.htm#au Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. Changes will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	None.

3.3.3 Local Conference Mixing

The phone's audio processing subsystem contains a flexible three-party conferencing capability[4]. This feature can be used to set up local three-party conferences where no external protocol signaling is involved.

3.3.4 Voice Activity Detection (VAD)

The purpose of VAD is to conserve network bandwidth by detecting periods of relative "silence" in the transmit data path and replacing that silence efficiently with special packets that indicate silence is occurring. For those compression algorithms without an inherent VAD function, such as G.711, the phone is compatible with the comprehensive codec-independent comfort noise transmission algorithm specified in RFC 3389. This algorithm is derived from G.711 Appendix II, which defines a comfort noise (CN) payload format (or bit-stream) for G.711 use in packet-based, multimedia communication systems. The phone generates CN packets (also known as Silence Insertion Descriptor (SID) frames) and also decodes CN packets, efficiently regenerating a facsimile of the background noise at the remote end.

Central (boot server)	Configuration file: sip.cfg	Enable or disable VAD and set the detection threshold. • For more information, see 4.6.1.8.10 Voice Activity Detection <VAD/> on page 102.

4. On SoundStation IP® 4000, conferences are not available if the G.729 codec is enabled on the phone. This restriction will be removed in future releases.

Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.3.5 DTMF Tone Generation

The phone generates DTMF tones in response to user dialing on the dial pad. These tones are transmitted in the RTP streams of connected calls. The phone can encode the DTMF tones using the active voice codec or using RFC 2833 compatible encoding. The coding format decision is based on the capabilities of the remote endpoint.

Central (boot server)	Configuration file: sip.cfg	Set the DTMF tone levels, autodialing on and off times, and other parameters. • For more information, see 4.6.1.5.1 Dual Tone Multi-Frequency <DTMF/> on page 83.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.3.6 DTMF Event RTP Payload

The phone is compatible with RFC 2833 - *RTP Payload for DTMF Digits, Telephony Tones, and Telephony Signals*. RFC 2833 describes a standard RTP-compatible technique for conveying DTMF dialing and other telephony events over an RTP media stream. The phone generates RFC 2833 (DTMF only) events but does not regenerate, nor otherwise use, DTMF events received from the remote end of the call.

Central (boot server)	Configuration file: sip.cfg	Enable or disable RFC 2833 support in SDP offers and specify the payload value to use in SDP offers. • For more information, see 4.6.1.5.1 Dual Tone Multi-Frequency <DTMF/> on page 83.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.3.7 Acoustic Echo Cancellation (AEC)

The phone employs advanced acoustic echo cancellation for hands-free operation. Both linear and non-linear techniques are employed to aggressively reduce echo yet provide for natural full-duplex communication patterns.

3.3.8 Audio Codecs

The following table summarizes the phone's audio codec support:

Algorithm	MIME Type	Ref.	Bit Rate	Sample Rate	Frame Size	Effective audio band-width
G.711μ-law	PMCU	RFC 1890	64 Kbps	8 Ksps	10ms - 80ms	3.5KHz
G.711a-law	PCMA	RFC 1890	64 Kbps	8 Ksps	10ms - 80ms	3.5KHz
G.729AB	G729	RFC 1890	8 Kbps	8 Ksps	10ms - 80ms	3.5KHz
SID	CN	RFC 3389	N/A	N/A	N/A	N/A
RFC 2833	phone-event	RFC 2833	N/A	N/A	N/A	N/A

Central (boot server)	Configuration file: sip.cfg	Specify codec priority, preferred payload sizes, and jitter buffer tuning parameters. For more information, see • 4.6.1.8.1.1 Codec Preferences <preferences/> on page 92, and • 4.6.1.8.1.2 Codec Profiles <profiles/> on page 93.
Local	Web Server (if enabled)	Specify codec priority, preferred payload sizes, and jitter buffer tuning parameters. Navigate to: http://<phoneIPAddress>/coreConf.htm#au Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. Changes will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	None.

3.3.9 Background Noise Suppression (BNS)

This feature, designed primarily for hands-free operation, reduces background noise to enhance communication in noisy environments.

3.3.10 Comfort Noise Fill

Comfort noise fill is designed to help provide a consistent noise level to the remote user of a hands-free call. Fluctuations in perceived background noise levels are an undesirable side effect of the non-linear component of most AEC systems. This feature uses noise synthesis techniques to smooth out the noise level in the direction toward the remote user, providing a more natural call experience.

3.3.11 Automatic Gain Control (AGC)

This feature, applicable to hands-free operation, is used to boost the transmit gain of the local talker in certain circumstances.[5] This increases the effective user-phone radius and helps with the intelligibility of soft-talkers.

3.4 Presence and Instant Messaging Features

The phone contains both Presence and Instant Messaging features. These features are compatible with Microsoft® Windows® Messenger and MSN® Messenger version 4.7 and Windows® Messenger 5.0. The phone's presence and instant messaging features are integrated with the contact directory features, using its contact database.

3.4.1 Presence

The Presence feature allows the phone to monitor the status of other users/devices and allows other users to monitor it. The status of monitored users is displayed visually and is updated in real time in the Buddies display screen or for speed dial entries on the phone's idle display. The user can block others from monitoring her phone and is notified when a change in monitored status occurs[6]. Phone status changes are broadcast automatically to monitoring phones when the user engages in calls or invokes do-

5. AGC support will be available in a subsequent release.
6. Notification when a change in monitored status occurs will be available in a subsequent release.

not-disturb. The user can also manually specify a state to convey, overriding, and perhaps masking, the automatic behavior.

Central (boot server)	XML file: <Ethernet address>-directory.xml	The <bw>0</bw> (buddy watching) and <bb>0</bb> (buddy blocking) elements in the <*Ethernet address*>-directory.xml file dictate the Presence aspects of directory entries. • For more information, see 3.1.17.1 Local Contact Directory File Format on page 31.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	The user can edit the directory contents. The *Watch Buddy* and *Block Buddy* fields control the buddy behavior of contacts. Changes will be stored in the phone's flash file system and backed up to the boot server copy of <*Ethernet address*>-directory.xml if this is configured. When the phone boots, the boot server copy of the directory, if present, will overwrite the local copy.

3.4.2 Instant Messaging

The phone supports sending and receiving instant text messages. The user is alerted to incoming messages visually and audibly. The user can choose to view the messages immediately or when it is convenient. For sending messages, the user can choose to either select a message from a pre-set list of short messages, or an alphanumeric text entry mode allows the typing of custom messages using the dial pad. Message sending can be initiated by replying to an incoming message or by initiating a new dialog. The destination for new dialog messages can be entered manually or selected from the contact directory, the preferred method.

3.5 Localization Features

3.5.1 Multilingual User Interface

All phones except SoundPoint® IP 300 and 301 have multilingual user interfaces. The System Administrator or the user can choose the language. Support for major western European languages is included and additional languages can be easily added. Support for Asian languages (Chinese, Japanese, and Korean) is also included but will render only on the SoundPoint® IP 600's and 601's and SoundStation® IP 4000's higher resolution displays.

Basic character support includes the following Unicode character ranges:

Name	Range
C0 Controls and Basic Latin	U+0000 - U+007F
C1 Controls and Latin-1 Supplement	U+0080 - U+00FF
Cyrillic (partial)	U+0400 - U+045F

Extended character support available on SoundPoint® IP 600 and SoundStation® IP 4000 platforms includes the following Unicode character ranges. Note that within a Unicode range, some characters may not be supported due to their infrequent usage.

Name	Range
CJK Symbols and Punctuation	U+3000 - U+303F
Hiragana	U+3040 - U+309F
Katakana	U+30A0 - U+30FF
Bopomofo	U+3100 - U+312F
Hangul Compatibility Jamo	U+3130 - U+318F
Bopomofo Extended	U+31A0 - U+31BF
Enclosed CJK Letters and Months	U+3200 - U+327F
CJK Compatibility	U+3300 - U+33FF
CJK Unified Ideographs	U+4E00 - U+9FFF
Hangul Syllables	U+AC00 - U+D7A3
CJK Compatibility Ideographs	U+F900 - U+FAFF
CJK Half-width forms	U+FF00 - U+FFFF

Note

The multilingual feature relies on dictionary files resident on the boot server. The dictionary files are downloaded from the boot server whenever the language is changed or at boot time when a language other than the internal US English language has been configured. If the dictionary files are inaccessible, the language will revert to the internal language.

Note
Currently, the multilingual feature is only available in the application. At this time, the bootROM application is English only.

Central (boot server)	Configuration file: sip.cfg	Specify the boot-up language and the selection of language choices to be made available to the user. For more information, see: • 4.6.1.3.1 Multilingual <multilingual/> on page 79, and • 4.6.1.3.1.1 Adding New Languages on page 80.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	The user can select the preferred language under the Settings menu. Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. Changes will permanently override global settings unless deleted via the Reset User Settings menu selection.

3.5.2 Downloadable Fonts

New fonts can be loaded onto the phone. For more information, see 4.6.1.14 Fonts <font/> on page 111.

3.5.3 Synthesized Call Progress Tones

In order to emulate the familiar and efficient audible call progress feedback generated by the PSTN and traditional PBX equipment, call progress tones are synthesized during the life cycle of a call. These call progress tones are easily configurable for compatibility with worldwide telephony standards or local preferences.

Central (boot server)	Configuration file: sip.cfg	Specify the basic tone frequencies, levels, and basic repetitive cadences. • For more information, see 4.6.1.5.2 Chord Sets <chord_sets/> on page 84 and 4.6.1.3.3 Call Progress Tones <callProgTones> on page 81. Specify downloaded sampled audio files for advanced call progress tones. • For more information, see 4.6.1.6 Sampled Audio for Sound Effects <sampled_audio/> on page 85. Specify patterns. For more information, see: • 4.6.1.7.1 Patterns <patterns/> on page 87, and • 4.6.1.7.1.1 Call Progress Patterns on page 89.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.6 Advanced Server Features

3.6.1 Voicemail Integration

The phone is compatible with voicemail servers. The subscribe contact and callback mode can be configured per user/registration on the phone. The phone can be configured with a SIP URL to be called automatically by the phone when the user elects to retrieve messages. Voicemail access can be configured to be via a single key press if only one registration has voicemail set up and the phone has a dedicated function key for this purpose (for example the Messages key on the SoundPoint® IP 500, 501, 600 and 601). A message-waiting signal from a voicemail server will trigger the message-waiting indicator to flash.

Central (boot server)	Configuration file: sip.cfg	For one-touch voicemail access, enable the “one-touch voicemail” user preference. • For more information, see 4.6.1.4 User Preferences <user_preferences/> on page 82.
	Configuration file: phone1.cfg	For one-touch voicemail access, choose to bypass instant messages to remove the step of selecting between instant messages and voicemail after pressing the Messages key on the SoundPoint® IP 500, 501, 600 and 601 (instant messages are still accessible from the Main Menu). On a per-registration basis, specify a subscribe contact for solicited NOTIFY applications, a callback mode (self call-back or another contact), and the contact to call when the user accesses voicemail. • For more information, see 4.6.2.5 Messaging <msg/> on page 137.
Local	Web Server (if enabled)	For one-touch voicemail access, enable the “one-touch voicemail” user preference and choose to bypass instant messages to remove the step of selecting between instant messages and voicemail after pressing the Messages key on the SoundPoint® IP 500, 501, 600 and 601 (instant messages are still accessible from the Main Menu). Navigate to: http://<phoneIPAddress>/coreConf.htm#us On a per-registration basis, specify a subscribe contact for solicited NOTIFY applications, a callback mode (self call-back or another contact) to call when the user accesses voicemail. Navigate to: http://<phoneIPAddress>/reg.htm Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. These changes will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	None.

3.6.2 Multiple Registrations

SoundPoint® IP phones support multiple registrations per phone and the SoundStation® IP 4000 supports a single registration. The SoundPoint® IP 300 and 301 support a maximum of two registrations, the SoundPoint® IP 500 and 501 support three and the SoundPoint® IP 600 and 601 support six. With the attachment of one or more

Expansion Modules, the SoundPoint® IP 601 supports an additional six registrations. A maximum of three Expansion Modules can be attached.

Each registration can be mapped to one or more line keys (a line key can be used for only one registration). The user can select which registration to use for outgoing calls or which to use when initiating new instant message dialogs.

Central (boot server)	Configuration file: sip.cfg	Specify the local SIP signaling port and an array of SIP servers to register to. For each server specify the registration period and the signaling failure behavior. • For more information, see 4.6.1.1.1 Local <local/> on page 71 and 4.6.1.1.2 Server <server/> on page 71.
	Configuration file: phone1.cfg	For up to twelve registrations, specify a display name, a SIP address, an optional display label, an authentication user ID and password, the number of line keys to use, and an optional array of registration servers. The authentication user ID and password are optional and for security reasons can be omitted from the configuration files. The local flash parameters will be used instead. The optional array of servers and their associated parameters will override the servers specified in sip.cfg if non-Null. • For more information, see 4.6.2.1 Registration <reg/> on page 128.

Local	Web Server (if enabled)	Specify the local SIP signaling port and an array of SIP servers to register to. Navigate to: http://<phoneIPAddress>/appConf.htm#se For up to six registrations (depending on the phone model, in this case the maximum is six even for the IP 601), specify a display name, a SIP address, an optional display label, an authentication user ID and password, the number of line keys to use, and an optional array of registration servers. The authentication user ID and password are optional and for security reasons can be omitted from the configuration files. The local flash parameters will be used instead. The optional array of servers will override the servers specified in sip.cfg in non-Null. This will also override the servers on the appConf.htm web page. Navigate to: http://<phoneIPAddress>/reg.htm Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. Changes will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	Use the SIP Configuration menu to specify the local SIP signaling port, a default SIP server to register to and registration information for up to twelve registrations (depending on the phone model). The SIP Configuration menu contains a sub-set of all the parameters available in the configuration files. Either the Web Server or the boot server configuration files or the local phone user interface should be used to configure registrations, not a mixture of these options. When the SIP Configuration menu is used, it is assumed that all registrations use the same server. Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. Changes will permanently override global settings unless deleted via the Reset User Settings menu selection. • For more information on the fields in this menu, see 4.6.1.1.1 Local <local/> on page 71, 4.6.1.1.2 Server <server/> on page 71 and 4.6.2.1 Registration <reg/> on page 128.

3.6.3 ACD login / logout

The phone allows ACD (Automatic Call Distribution) login and logout. This feature depends on support from a SIP server.

Central (boot server)	Configuration file: sip.cfg	Turn this feature on or off. • For more information, see 4.6.1.23 Feature <feature/> on page 125.
	Configuration file: phone1.cfg	Enable this feature per registration. • For more information, see 4.6.2.1 Registration <reg/> on page 128.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.6.4 ACD agent available / unavailable

The phone supports ACD (Automatic Call Distribution) agent available and unavailable. This feature depends on support from a SIP server.

Central (boot server)	Configuration file: sip.cfg	Turn this feature on or off. • For more information, see 4.6.1.23 Feature <feature/> on page 125.
	Configuration file: phone1.cfg	Enable this feature per registration. • For more information, see 4.6.2.1 Registration <reg/> on page 128.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

3.6.5 Server Redundancy

The phone can be configured with multiple SIP servers, one primary and one or more backup. The phone will switch to a backup server when the current primary server fails. Backup server configuration can be static or can use advanced DNS methods. In the case of static server lists, when a server registration fails, registration will be attempted on another server. If the phone is not registered to the first server in the list when registration fails, it will start by trying to register to the first server. When making a new call, if the INVITE fails, the other servers in the list will be tried one by one for routing signaling until the last server is tried.

Definition of signaling failure (registration or start of call):

- If TCP is used: The signaling fails if the connection fails or the Send fails.
- If UDP is used: The signaling fails if ICMP is detected or if the signal times out. If the signaling has been attempted via all servers in the list and this is the last server then the signaling fails after the complete UDP timeout defined in RFC 3261. If it is not the last server in the list, the maximum number of retries using the configurable retry timeout is used. For more information, see 4.6.1.1.2 Server <server/> on page 71 and 4.6.2.1 Registration <reg/> on page 128.

3.6.5.1 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 - *Locating SIP Servers*. If a port is given, the only lookup will be an A record. If no port is given, NAPTR and SRV records will be tried, before falling back on A records if NAPTR and SRV records return no results. If no port is given, and none is found through DNS, 5060 will be used.

See *http://www.ietf.org/rfc/rfc3263.txt* for an example.

3.7 Accessory Internet Features

3.7.1 MicroBrowser

The SoundPoint® IP 600 phone supports an XHTML microbrowser. This can be launched by pressing the Services key.

Central (boot server)	Configuration file: sip.cfg	Specify the Services browser home page, a proxy to use, and size limits. • For more information, see 4.6.1.25 MicroBrowser <microbrowser/> on page 127.

Local	Web Server (if enabled)	Specify the Services browser home page and proxy to use. Navigate to: http://<phoneIPAddress>/coreConf.htm#mb Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. Changes will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	None

3.8 Security Features

3.8.1 Local User and Administrator Privilege Levels

Several local settings menus are protected with two privilege levels, user and administrator, each with its own password. The phone will prompt for either the user or administrator password before granting access to the various menu options. When the user password is requested, the administrator password will also work. The web server is protected by the administrator password.

Central (boot server)	Configuration file: sip.cfg	Specify the minimum lengths for the user and administrator passwords. • For more information, see 4.6.1.19.1 Password Lengths <pwd/><length/> on page 122.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	The user and administrator passwords can be changed under the Settings menu. Passwords can consist of ASCII characters 32-127 (0x20-0x7F) only. Changes are saved to local flash but are not backed up to <*Ethernet address*>-phone.cfg on the boot server for security reasons.

3.8.2 Custom Certificates

When trying to establish a connection to a boot server for application provisioning, the phone trusts certificates issued by widely recognized certificate authorities. See 6.1 Trusted Certificate Authority List on page 151. In addition, custom certificates can be added to the phone. This is done by using the SSL Security menu on the phone to provide the URL of the custom certificate then select an option to use this custom certificate.

Central (boot server)	Configuration file:	None.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	The custom certificate can be specified and the type of certificate to trust can be set under the Settings menu.

3.8.3 Incoming Signaling Validation

Three optional levels of security are provided for validating incoming network signaling:

- source IP address validation
- digest authentication
- both

Central (boot server)	Configuration File: sip.cfg	Specify the type of validation to perform on a request-by-request basis, appropriate to specific event types in some cases. • For more information, see 4.6.1.1.3.3 Request Validation <requestValidation/> on page 75.
Local	Web Server (if enabled)	None.
	Local Phone User Interface	None.

4 Optimization

4.1 Ethernet Switch

The SoundPoint® IP phones contain two Ethernet ports, labeled LAN and PC, and an embedded Ethernet switch that runs at full line-rate. The Ethernet switch allows a personal computer and other Ethernet devices to connect to the office LAN by daisy chaining through the phone, eliminating the need for a stand-alone hub. The SoundPoint® IP switch gives higher transmit priority to packets originating in the phone. SoundPoint® IP can be powered via a local AC power adapter or can be line-powered (power supplied via the signaling or idle pairs of the LAN Ethernet cable). Line powering typically requires that the phone plugs directly into a dedicated LAN jack. Devices that do not require LAN power can then plug into the SoundPoint® IP PC Ethernet port.

SoundPoint® IP Switch - Port Priorities

To help ensure good voice quality, the Ethernet switch embedded in the SoundPoint® IP phones should be configured to give voice traffic emanating from the phone higher transmit priority than those from a device connected to the PC port. If not using a VLAN (VLAN blank in the setup menu), this will automatically be the case. If using a VLAN, ensure that the 802.1p priorities for both default and RTP packet types are set to 2 or greater. Otherwise, these packets will compete equally with those from the PC port. For more information, see 4.6.1.9 Quality of Service <QOS/> on page 102.

4.2 Application Network Setup

4.2.1 RTP Ports

The phone is compatible with RFC 1889 - *RTP: A Transport Protocol for Real-Time Applications* - and the updated RFCs 3550 and 3551. Consistent with RFC 1889, the phone treats all RTP streams as bi-directional from a control perspective and expects that both RTP endpoints will negotiate the respective destination IP addresses and ports. This allows RTCP to operate correctly even with RTP media flowing in only a single direction, or not at all. It also allows greater security: packets from unauthorized sources can be rejected.

The phone can filter incoming RTP packets arriving on a particular port by IP address. Packets arriving from a non-negotiated IP address can be discarded.

The phone can also enforce symmetric port operation for RTP packets: packets arriving with the source port set to other than the negotiated remote sink port can be rejected.

The phone can also jam the destination transport port to a specified value regardless of the negotiated port. This can be useful for punching through firewalls. When this is enabled, all RTP traffic will be sent to the specified port and will be expected to arrive on that port as well. Incoming packets are sorted by the source IP address and port, allowing multiple RTP streams to be multiplexed.

The RTP port range used by the phone can be specified. Since conferencing and multiple RTP streams are supported, several ports can be used concurrently. Consistent with RFC 1889, the next higher odd port is used to send and receive RTCP.

Central (boot server)	Configuration file: sip.cfg	Specify whether to filter incoming RTP packets by IP address, whether to require symmetric port usage, whether to jam the destination port and specify the local RTP port range start. • For more information, see 4.6.1.10.3.1 RTP <RTP/> on page 106.
Local	Web Server (if enabled)	Specify whether to filter incoming RTP packets by IP address, whether to require symmetric port usage, whether to jam the destination port and specify the local RTP port range start. Navigate to: http://<phoneIPAddress>/netConf.htm#rt Changes are saved to local flash and backed up to <*Ethernet address*>-phone.cfg on the boot server. They will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	None.

4.2.2 Working with Network Address Translation (NAT)

The phone can work with certain types of network address translation (NAT). The phone's signaling and RTP traffic use symmetric ports (the source port in transmitted packets is the same as the associated listening port used to receive packets) and the external IP address and ports used by the NAT on the phone's behalf can be configured on a per-phone basis.

Central (boot server)	Configuration file: phone1.cfg	Specify the external NAT IP address and the ports to be used for signaling and RTP traffic. • For more information, see 4.6.2.6 Network Address Translation <nat/> on page 138.
Local	Web Server (if enabled)	Specify the external NAT IP address and the ports to be used for signaling and the RTP traffic. Navigate to: http://<phoneIPAddress>/netConf.htm#na Changes are saved to local flash and backed up to *<Ethernet address>*-phone.cfg on the boot server. Changes will permanently override global settings unless deleted via the Reset User Settings menu selection.
	Local Phone User Interface	None.

4.3 Updating and Rebooting

The bootROM, application executable, and configuration files can be updated automatically via the centralized provisioning (boot server) model.

To automatically update:

1. Back up old application and configuration files. The old configuration can be easily restored by reverting to the back-up files.
2. Customize new configuration files or apply new or changed parameters to the old configuration files. Differences between old and new versions of configuration files are explained in the Release Notes which accompany the software. Changes to site-wide configuration files such as sip.cfg can be done manually, but a scripting tool is useful to change per-phone configuration files.
3. Save the new configuration files and images (such as sip.ld) on the boot server.
4. Reboot the phones. See Manual Reboot: Menu Option or Key Presses on page 65.

For more information, see 2.2.2 Application Configuration on page 11.

Manual Reboot: Menu Option or Key Presses

To reboot phones manually, a menu option can be selected or a key combination can be used. The menu option is called Restart Phone and it is found in the Settings menu.

For the key combination, press and hold the following keys simultaneously until a confirmation tone is heard or for about three seconds:

SoundPoint® IP 300 and 301:	Volume-, Volume+, Hold, Do Not Disturb
SoundPoint® IP 500 and 501:	Volume-, Volume+, Hold, Messages
SoundPoint® IP 600 and 601:	Volume-, Volume+, Mute, Messages
SoundStation® IP 4000:	*, #, Volume+, Select

Centralized Reboot

The phones can be rebooted remotely via the SIP signaling protocol. Refer to 4.6.1.1.3.4 Special Events <specialEvent/> on page 76.

Periodic Polling For Upgrades

The phones can be configured to periodically poll the boot server to check for changed configuration files or application executable. If a change is detected the phone will reboot to download the change. Refer to 4.6.1.20 Provisioning <provisioning/> on page 123.

4.4 Event Logging

The phones maintain both boot and application event log files. These files can be helpful when diagnosing problems. The event log files are stored in the phone's flash file system and are periodically uploaded to the provisioning boot server if permitted by security policy. The files are stored in the phone's home directory or a user-configurable directory on the boot server. Both overwrite and append[7] modes are supported for the application event log.

The event log files are:

- <*Ethernet address*>-boot.log
- <*Ethernet address*>-app.log

The boot log file is uploaded to the boot server after every reboot.

The application log file is uploaded periodically or when the local copy reaches a predetermined size.

7. Note: HTTP and TFTP don't support append mode unless server settings are changed for this.

As an additional diagnostic tool, both log files can be uploaded on demand to the boot server by pressing and holding the following keys until a confirmation tone is heard or for about three seconds:

SoundPoint® IP 300 and 301:	Line1, Line2, Arrow Up, Arrow Down
SoundPoint® IP 500 and 501:	The four arrow keys.
SoundPoint® IP 600 and 601:	The four arrow keys.
SoundStation® IP 4000:	Menu, Exit, Off-hook/Hands-free, Redial

Log files uploaded in this manner are named:

- <*Ethernet address*>-now-boot.log
- <*Ethernet address*>-now-app.log

Central (boot server)	Configuration file: sip.cfg	Specify a multitude of event logging settings. • For more information, see 4.6.1.18 Event Logging <logging/> on page 119.
	Configuration file: <*Ethernet address*>.cfg	Specify different directory to use for log files if desired. • For more information, see 2.2.2.1.1.1 Master Configuration Files on page 12.
Local	Web Server (if enabled)	Specify a multitude of event logging settings. Navigate to: http://<phoneIPAddress>/coreConf.htm#lo
	Local Phone User Interface	None.

4.5 Audio Quality Issues and VLANs

The phone contains both network layer and Ethernet layer support for prioritizing voice and signaling traffic over the network. Quality of Service (QoS) parameters include IP type-of-service (TOS) bits, and Ethernet IEEE 802.1p user priority. These can be set on a per-protocol basis. The phone also supports RTCP per RFC 1889.

4.5.1 IP TOS

The “type of service” field in an IP packet header consists of four TOS bits and a 3-bit precedence field. Each TOS bit can be set to either 0 or 1. The precedence field can be set to a value from 0 through 7. The type of service can be configured specifically for RTP packets and call control packets, such as SIP signaling packets.

Central (boot server)	Configuration file: sip.cfg	Specify protocol-specific IP TOS settings. • For more information, see 4.6.1.9.2 IP TOS <IP/> on page 103.
Local	Web Server (if enabled)	Specify IP TOS settings. Navigate to: http://<phoneIPAddress>/netConf.htm#qo
	Local Phone User Interface	None.

4.5.2 IEEE 802.1p/Q

The phone will tag all Ethernet packets it transmits with an 802.1Q VLAN header:

1. when it has a valid VLAN ID set in its network configuration, or
2. is instructed to tag packets via Cisco Discovery Protocol (CDP) running on a connected Ethernet switch, or
3. a VLAN ID is obtained from DHCP (see 2.2.1.3.2 DHCP Menu on page 8).

The 802.1p/Q user_priority field can be set to a value from 0 to 7. The user_priority can be configured specifically for RTP packets and call control packets, such as SIP signaling packets, with default settings configurable for all other packets.

Central (boot server)	Configuration file: sip.cfg	Specify default and protocol-specific 802.1p/Q settings. • For more information, see 4.6.1.9.1 Ethernet IEEE 802.1p/Q <Ethernet/> on page 102.
Local	Web Server (if enabled)	Specify 802.1p/Q settings. Navigate to http://<phoneIPAddress>/netConf.htm#qo
	Local Phone User Interface	Specify whether CDP is to be used or manually set the VLAN ID or configure DHCP VLAN Discovery. Phase 1: bootRom - Navigate to: SETUP menu during auto-boot countdown. Phase 2: Application - Navigate to: Menu>Settings>Advanced>Admin Settings>Network Configuration • For more information, see 2.2.1 Basic Network Setup on page 4.

4.5.3 RTCP Support

The phone supports RTCP per RFC 1889. For each RTP stream, which, by convention, uses even ports only, the next higher odd port is used to send and receive RTCP reports.

4.6 Configuration Files

This section is a reference for all parameters that are configurable when using the centralized provisioning installation model. It is divided into two sections:

- Application Configuration - sip.cfg
- Per-phone Configuration - phone1.cfg

Notes

In the following tables, “Null” should be interpreted as the empty string, that is, attributeName=“” when the file is viewed in a text editor.

To enter special characters in a configuration file, enter the appropriate sequence using a **text editor**. See the following table.

Special Character	Required Character Sequence in Text Editor
&	&
”	"
’	'
<	<
>	>

4.6.1 SIP Configuration - sip.cfg

The configuration file sip.cfg contains SIP protocol and core configuration settings that would typically apply to an entire installation and must be set before the phones will be operational, unless changed via the local web server interface or local menu settings on the phone. Settings include the local port used for SIP signaling, the address and ports of a cluster of SIP servers, and other parameters. The following sections describe each of these parameters.

4.6.1.1 Protocol <voIpProt/>

4.6.1.1.1 Local <local/>

Attribute	Permitted Values	Default	Interpretation
volpProt.local.port	0 to 65535	5060	Local port for sending and receiving SIP signaling packets. If set to 0 or Null, 5060 is used for the local port but it 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.

4.6.1.1.2 Server <server/>

<table>
<tr><th>Attribute</th><th>Permitted Values</th><th>Default</th><th>Interpretation</th></tr>
<tr><td>voIpProt.server.x.address</td><td>dotted-decimal IP address or host name</td><td>Null</td><td rowspan="2">IP address or host name and port of a SIP server that accepts registrations. Multiple servers can be listed starting with x=1, 2, ... for fault tolerance.
If port is 0 or Null:
If voIpProt.server.x.address is a hostname and voIp-Prot.server.x.transport is set to DNSnaptr, do NAPTR then SRV lookups.
If voIpProt.server.x.transport is set to TCPpreferred or UDPonly then use 5060 and don't advertise the port number in signalling.
If voIpProt.server.x.address is an IP address, there is no DNS lookup and 5060 is used for the port but it is not advertised in signaling.
If port is 1 to 65535:
This value is used and it is advertised in signaling.</td></tr>
<tr><td>voIpProt.server.x.port</td><td>0, Null, 1 to 65535</td><td>Null</td></tr>
</table>

Attribute	Permitted Values	Default	Interpretation
voIpProt.server.x.transport	DNSnaptr or TCPpreferred or UDPonly	DNSnaptr	If set to Null or DNSnaptr: If voIpProt.server.x.address is a hostname and voIp-Prot.server.x.port is 0 or Null, do NAPTR then SRV look-ups to try to discover the transport, ports and servers, as per RFC 3263. If voIp-Prot.server.x.address is an IP address, or a port is given, then UDP is used. If set to TCPpreferred: TCP is the preferred transport, UDP is used if TCP fails. If set to UDPonly: Only UDP will be used.
voIpProt.server.x.expires	positive integer, minimum 300	3600	Requested registration period in seconds[a].
voIpProt.server.x.register	0, 1	1	If set to 0, calls can be routed to an outbound proxy without reg-istration.
voIpProt.server.x.retryTimeOut	Null or non-negative integer	0	If set to 0 or Null, use standard RFC 3261 signaling retry behavior. Otherwise retryTim-eOut determines how often retries will be sent. Units = milliSeconds. (Finest resolution = 100ms).
voIpProt.server.x.retryMaxCount	Null or non-negative integer	3	If set to 0 or Null, 3 is used. retryMaxCount retries will be attempted before moving on to the next available server.
voIpProt.server.x.expires.lineSeize	positive integer, minimum 10	30	Requested line-seize subscrip-tion period.

a. This is the phone's requested registration period. The period negotiated with the server may be different. The phone will attempt to re-register when half the negotiated period has expired.

4.6.1.1.3 SIP <SIP/>

Attribute	Permitted Values	Default	Interpretation
voIpProt.SIP.useRFC2543hold	0, 1	0	If set to 1, use the obsolete c=0.0.0.0 RFC2543 technique, otherwise, use SDP media direction attributes (such as a=sendonly) per RFC 3264 when initiating hold. In either case, the phone processes incoming hold signaling in either format.
voIpProt.SIP.lcs	0, 1	0	If set to 1, the proprietary "epid" parameter is added to the From field of all requests to support Windows Live Communications Server.
voIpProt.SIP.sendCompactHdrs	0, 1	0	If set to 0, SIP header names generated by the phone use the long form, for example 'From'. If set to 1, SIP header names generated by the phone use the short form, for example 'f'.
voIpProt.SIP.WM50	0, 1	0	If set to 1, Windows Messenger® 5.0 will be supported. If set to 0, Windows Messenger® 4.7 will be supported.
voIpProt.SIP.keepalive.session-Timers	0, 1	0	If set to 1, the session timer will be enabled. If set to 0, the session timer will be disabled, and the phone will not declare "timer" in "Support" header in INVITE. The phone will still respond to a re-INVITE or UPDATE. The phone will not try to re-INVITE or do UPDATE even if remote endpoint asks for it.
voIpProt.SIP.request-URI.E164.addGlobalPrefix	0, 1	0	If set to 1, '+' global prefix is added to E.164 user parts in sip: URIs:.
voIpProt.SIP.allowTransferOn-Proceeding	0, 1	1	If set to 1, a transfer can be completed during the proceeding state of a consultation call. This is the default. If set to 0, a transfer is not allowed during the proceeding state of a consultation call.

4.6.1.1.3.1 Outbound Proxy <outboundProxy/>

Attribute	Permitted Values	Default	Interpretation
voIpProt.SIP.outboundProxy.address	dotted-decimal IP address or host name	Null	IP address or host name and port of a SIP server to which the phone shall send all requests.
voIpProt.SIP.outboundProxy.port	1 to 65535	5060	
voIpProt.SIP.outboundProxy.transport	DNSnaptr or TCPpreferred or UDPonly	DNSnaptr	If set to Null or DNSnaptr: If voIpProt.SIP.outboundProxy.address is a hostname and voIpProt.SIP.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 voIpProt.SIP.outboundProxy.address is an IP address, or a port is given, then UDP is used. If set to TCPpreferred: TCP is the preferred transport, UDP is used if TCP fails. If set to UDPonly: Only UDP will be used.

4.6.1.1.3.2 Alert Information <alertInfo/>

Attribute	Permitted Values	Default	Interpretation
volpProt.SIP.alertInfo.x.value	string to compare against the value of Alert-Info headers in INVITE requests	Null	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 (see 4.6.1.7.2 Ring type <ringType/> on page 91) will be applied.
voIpProt.SIP.alertInfo.x.class	positive integer	Null	

4.6.1.1.3.3 Request Validation <requestValidation/>

Attribute	Permitted Values	Default	Interpretation
voIpProt.SIP.requestValidation.x.request	One of: "INVITE", "ACK", "BYE", "REGISTER", "CANCEL", "OPTIONS", "INFO", "MESSAGE", "SUBSCRIBE", "NOTIFY", "REFER", "PRACK", or "UPDATE"	Null	Sets the name of the method for which validation will be applied[a].
voIpProt.SIP.requestValidation.x.method	Null or one of: "source", "digest" or "both"/"all"	Null	If Null, no validation is done. 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
voIpProt.SIP.requestValidation.x.request.y.event	A valid string	Null	Determines which events specified with the Event header should be validated; only applicable when voIpProt.SIP.requestValidation.x.request is set to "SUBSCRIBE" or "NOTIFY". If set to Null, all events will be validated.
voIpProt.SIP.requestValidation.digest.realm	A valid string	PolycomSPIP	Determines string used for Realm.

a. WARNING: Intensive request validation may have a negative performance impact due to the additional signaling required in some cases, therefore, use it judiciously.

4.6.1.1.3.4 Special Events <specialEvent/>

Attribute	Permitted Values	Default	Interpretation
voIpProt.SIP.specialEvent.lineSeize.nonStandard	0, 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.
voIpProt.SIP.specialEvent.checkSync.alwaysReboot	0, 1	0	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 [mac].cfg have changed on the FTP server when a NOTIFY message is received from the server with event equal to check-sync.

4.6.1.1.3.5 Conference Setup <conference/>

Attribute	Permitted Values	Default	Interpretation
voIpProt.SIP.conference.address	ASCII string up to 128 characters long	Null	If Null, conferences are set up on the phone locally. If set to some value, conferences are set up by the server using the conferencing agent specified by this address. The acceptable values depend on the conferencing server implementation policy.

4.6.1.2 Dial Plan <dialplan/>

Attribute	Permitted Values	Default	Interpretation
dialplan.impossibleMatch-Handling	0, 1 or 2	0	If set to 0, the digits entered up to and including the point where an impossible match occurred are sent to the server immediately. If set to 1, give reorder tone. If set to 2, allow user to accumulate digits and dispatch call manually with the Send soft key.
dialplan.removeEndOfDial	0, 1	1	If set to 1, strip trailing # digit from digits sent out.

4.6.1.2.1 Digit Map <digitmap/>

Attribute	Permitted Values	Default	Interpretation
dialplan.digitmap	string compatible with the digit map feature of MGCP described in 2.1.5 of RFC 3435. String is limited to 512 bytes and 20 segments; a comma is also allowed; when reached in the digit map, a comma will turn dial tone back on.	[2-9]11|0T|011xxx.T|[0-1][2-9]xxxxxxxxx|[2-9]xxxxxxxxx|[2-9]xxxT	When this attribute is present, number-only dialing during the setup phase of new calls will be compared against the patterns therein and if a match is found, the call will be initiated automatically eliminating the need to press Send.
dialplan.digitmap.timeOut	positive integer	3	Timeout in seconds for 'T' feature of digitmap.

4.6.1.2.2 Routing <routing/>

This configuration section allows the user to create a specific routing path for outgoing SIP calls independent of other 'default' configuration.

4.6.1.2.2.1 Server <server/>

Attribute	Permitted Values	Default	Interpretation
dialplan.routing.server.x.address	dotted-decimal IP address or host name	Null	IP address or host name and port of a SIP server that will be used for routing calls. Multiple servers can be listed starting with x=1, 2, ... for fault tolerance.
dialplan.routing.server.x.port	1 to 65535	5060	

4.6.1.2.2.2 Emergency <emergency/>

In the following attributes, *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.

Attribute	Permitted Values	Default	Interpretation
dialplan.routing.emergency.x.value	Comma separated list of entries or single entry representing a SIP URL or a combination of SIP URLs.	Null Example: “15,17,18”, “911”, “sos”.	This determines the URLs that should be watched for. When one of these defined URLs is detected as having been dialed by the user, the call will automatically be directed to the defined emergency server.
dialplan.routing.emergency.x.server.y	positive integer	Null	Index representing the server defined in 4.6.1.2.2.1 Server <server/> on page 78 that will be used for emergency routing.

4.6.1.3 Localization <localization/>

The phone has a multilingual user interface. It supports both North American and international time and date formats. The call progress tones can also be customized. For more information, see 4.6.1.3.3 Call Progress Tones <callProgTones> on page 81, 4.6.1.5.2 Chord Sets <chord_sets/> on page 84, and 4.6.1.7.1.1 Call Progress Patterns on page 89.

4.6.1.3.1 Multilingual <multilingual/>

The multilingual feature is based on string dictionary files downloaded from the boot server. These files are encoded in standalone XML format. Several western European and Asian languages are included with the distribution.

Attribute	Permitted Values	Interpretation
lcl.ml.lang	Null OR An exact match for one of the folder names under the SoundPointIPLocalization folder on the boot server.	If Null, the default internal language (US English) will be used, otherwise, the language to be used may be specified in the format *language-region.*
lcl.ml.lang.menu.x	String in the format *language_region*	Multiple lcl.ml.lang.menu.x attributes are supported - as many languages as are desired. However, the lcl.ml.lang.menu.x attributes must be sequential (lcl.ml.lang.menu.1, lcl.ml.lang.menu.2, lcl.ml.lang.menu.3, ..., lcl.ml.lang.menu.N) with no gaps and the strings must exactly match a folder name under the SoundPointIPLocalization folder on the boot server for the phone to be able to locate the dictionary file.
lcl.ml.lang.cpt.x	positive integer	The call progress tone index to be associated with this language. See 4.6.1.3.3 Call Progress Tones <callProgTones> on page 81.
lcl.ml.lang.clock.x.24HourClock	0,1	If attribute present, overrides lcl.datetime.time.24HourClock; If 1, display time in 24-hour clock mode rather than am/pm.

Attribute	Permitted Values	Interpretation
lcl.ml.lang.clock.x.format	string which includes ‘D’, ‘d’ and ‘M’ and two optional commas	If attribute 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, 1	If attribute 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.clock.x.dateTop	0, 1	If attribute present, overrides lcl.datetime.date.dateTop; If 1, display date above time, otherwise display time above date.
lcl.ml.lang.y.list	“All” or a comma-separated list	A list of the languages supported on hardware platform ‘y’ where ‘y’ can be IP_500 or IP_600.

4.6.1.3.1.1 Adding New Languages

Follow these steps to add new languages to those included with the distribution:

1. Create a new dictionary file based on an existing one.
2. Change the strings making sure to encode the XML file in UTF-8 but also ensuring the UTF-8 characters chosen are within the Unicode character ranges indicated in 3.5.1 Multilingual User Interface on page 52.
3. Place the file in an appropriately named folder according to the format *language_region* parallel to the other dictionary files under the SoundPoint-IPLocalization folder on the boot server.
4. Add a lcl.ml.lang.clock.menu.x attribute to the configuration file.
5. Add lcl.ml.lang.clock.x.24HourClock, lcl.ml.lang.clock.x.format, lcl.ml.lang.clock.x.longFormat and lcl.ml.lang.clock.x.dateTop attributes and set them according to the regional preferences.
6. (Optional) Set lcl.ml.lang to be the new *language_region* string.

4.6.1.3.2 Date and Time <datetime/>

Attribute	Permitted Values	Interpretation
lcl.datetime.time.24HourClock	0,1	If 1, display time in 24-hour clock mode rather than a.m./p.m.
lcl.datetime.date.format	string which includes 'D', 'd' and 'M' and two optional commas	Controls format of date string. D = day of week d = day M = month Up to two commas may be included. For example: 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	0,1	If 1, display the day and month in long format (Friday/November), otherwise, use abbreviations (Fri/Nov).
lcl.datetime.date.dateTop	0, 1	If 1, display date above time else display time above date.

4.6.1.3.3 Call Progress Tones <callProgTones>

Call progress tone overrides can be used to customize the tones for a particular country or region. The overrides set offered by default spans all default languages on the phone. Tone overrides are based on the ITU-T Recommendation E.180 Supplement 2 entitled *Telephone Network and ISDN - Operation, numbering, routing and mobile service - Various tones used in national networks*.

Attribute	Permitted Values	Interpretation
lcl.cpt	positive integer OR blank	The index of the default tone overrides to be selected by the phone. If blank, default call progress tones are used.
lcl.cpt.menu.x	string	String to specify the country or region such as Italy. Multiple lcl.cpt.menu.x strings are supported, the strings are displayed in the Call Progress Tones menu. The lcl.cpt.menu.x attributes must be sequential (lcl.cpt.menu.1, lcl.cpt.menu.2, lcl.cpt.menu.3, ..., lcl.cpt.menu.N) with no gaps.

In the following table, *x* is the index of the region as specified by the *x* index of the lcl.cpt.menu.x attribute above, *y* is the chord set number and *cat* is one of `cp` or `misc`. For more information, see 4.6.1.7.1.1 Call Progress Patterns on page 89.

Attribute	Permitted Values	Interpretation
lcl.cpt.chord.*cat*.x.y.freq.z	0-1600	Frequency for this component in Hertz; up to four chord-set components can be specified (z=1, 2, 3, 4).
lcl.cpt.chord.*cat*.x.y.level.z	-57 to 3	Level of this component in dBm0.
lcl.cpt.chord.*cat*.x.y.onDur	positive integer	On duration in milliseconds, 0=infinite.
lcl.cpt.chord.*cat*.x.y.offDur	positive integer	Off duration in milliseconds, 0=infinite.
lcl.cpt.chord.*cat*.x.y.repeat	positive integer	Specifies how many times the ON/OFF cadence is repeated, 0=infinite.

4.6.1.4 User Preferences <user_preferences/>

Attribute	Permitted Values	Default	Interpretation
up.headsetMode	0,1	0	If set to 1, the headset will be selected as the preferred transducer after its first use until the headset key is pressed again; otherwise, hands-free will be selected preferentially over the headset.
up.useDirectoryNames	0,1	0	If set to 1, the name fields of directory entries which match incoming calls will be used for caller identification display and in the call lists instead of the name provided via network signaling.
up.oneTouchVoiceMail	0, 1	0	If set to 1, the voicemail summary display is bypassed and voicemail is dialed directly (if configured).
up.welcomeSoundEnabled	0, 1	1	If set to 1, play welcome sound effect after a reboot.
up.welcomeSoundOnWarmBootEnabled	0, 1	0	If set to 1, play welcome sound effect on warm as well as cold boots, otherwise only a cold boot will trigger the welcome sound effect.

Attribute	Permitted Values	Default	Interpretation
up.localClockEnabled	0, 1	1	If set to 1, display the date and time on the idle display

4.6.1.5 Tones <tones/>

This section describes configuration items for the tone resources available in the phone.

4.6.1.5.1 Dual Tone Multi-Frequency <DTMF/>

Attribute	Permitted Values	Default	Interpretation
tone.dtmf.level	-33 to -3	-15	Level of the high frequency component of the DTMF digit measured in dBm0; the low frequency tone will be two dB lower.
tone.dtmf.onTime	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 generated for; this is also the minimum time the tone will be played for when dialing manually (even if key press is shorter).
tone.dtmf.offTime	positive integer	50	When a sequence of DTMF tones is played out automatically, this is the length of time in milliseconds the phone will pause between digits; this is also the minimum inter-digit time when dialing manually.
tone.dtmf.chassis.masking	0, 1	0	If set to 1, DTMF tones will be substituted with a non-DTMF pacifier tone when dialing in hands-free mode. This prevents DTMF digits being broadcast to other surrounding telephony devices or being inadvertently transmitted in-band due to local acoustic echo. Note: tone.dtmf.chassis.masking should only be enabled when tone.dtmf.viaRtp is disabled.

Attribute	Permitted Values	Default	Interpretation
tone.dtmf.stim.pac.offHookOnly	0, 1	0	Not currently used.
tone.dtmf.viaRtp	0, 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. Note: tone.dtmf.chassis.masking should be enabled when tone.dtmf.viaRtp is disabled.
tone.dtmf.rfc2833Control	0, 1	1	If set to 1, the phone will indicate a preference for encoding DTMF via RFC 2833 format in its Session Description Protocol (SDP) offers by showing support for the phone-event payload type; this does not affect SDP answers, these will always honor the DTMF format present in the offer since the phone has native support for RFC 2833.
tone.dtmf.rfc2833Payload	96-127	101	The phone-event payload encoding in the dynamic range to be used in SDP offers.

4.6.1.5.2 Chord Sets <chord_sets/>

Chord sets are the building blocks of sound effects that use synthesized rather than sampled audio (most call progress and ringer sound effects). 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 blocks of chord sets:

- callProg: used for call progress sound effect patterns
- ringer
- misc (miscellaneous)

All three blocks use the same chord set specification format.

In the following table, *x* is the chord-set number and *cat* is one of `callProg`, `ringer`, or `misc`.

Attribute	Permitted Values	Interpretation
tone.chord.*cat*.x.freq.y	0-1600	Frequency for this component in Hertz; up to four chord-set components can be specified (y=1, 2, 3, 4).
tone.chord.*cat*.x.level.y	-57 to 3	Level of this component in dBm0.
tone.chord.*cat*.x.onDur	positive integer	On duration in milliseconds, 0=infinite.
tone.chord.*cat*.x.offDur	positive integer	Off duration in milliseconds, 0=infinite.
tone.chord.*cat*.x.repeat	positive integer	Specifies how many times the ON/OFF cadence is repeated, 0=infinite.

4.6.1.6 Sampled Audio for Sound Effects <sampled_audio/>

The following sampled audio WAVE file (.wav) formats are supported:

- mono 8 kHz G.711 µ-Law
- G.711 A-Law
- L16/16000[8] (16-bit, 16 kHz sampling rate, mono)

The phone uses built-in wave files for some sound effects. The built-in wave files can be replaced with files downloaded from the boot server or from the Internet, however, these are stored in volatile memory so the files will need to remain accessible should the phone need to be rebooted. Files will be truncated to a maximum size of 300 kilobytes.

8. L16/16000 is not supported on SoundPoint® IP 300, 301 and SoundStation® IP 4000 phones.

In the following table, *x* is the sampled audio file number.

Attribute	Permitted Values	Interpretation
saf.x	Null OR valid path name OR an RFC 1738-compliant URL to a HTTP, FTP, or TFTP wave file resource. Note: Refer to the above wave file format restrictions.	If Null, the phone will use a built-in file. If set to a path name, the phone will attempt to download this file at boot time from the boot server. If set to a URL, the phone will attempt to download this file at boot time from the Internet. Note: A TFTP URL is expected to be in the format: tftp://<host>/[pathname]<filename>, for example: tftp://somehost.example.com/sounds/example.wav

The following table defines the default usage of the sampled audio files with the phone:

Sampled Audio File	Default use within phone (pattern reference)
1	Welcome Sound Effect (se.pat.misc.7)
2	Ringer 13 (se.pat.ringer.13)
3	Ringer 14 (se.pat.ringer.14)
4	Ringer 15 (se.pat.ringer.15)
5	Ringer 16 (se.pat.ringer.16)
6	Ringer 17 (se.pat.ringer.17)
7	Ringer 18 (se.pat.ringer.18)
8	Ringer 19 (se.pat.ringer.19)
9	Ringer 20 (se.pat.ringer.20)
10	Ringer 21 (se.pat.ringer.21)
11	Ringer 22 (se.pat.ringer.22)
12-24	Not used.

4.6.1.7 Sound Effects <sound_effects/>

The phone uses both synthesized (based on the chord-sets described earlier) and sampled audio sound effects. Sound effects are defined by patterns: rudimentary sequences of chord-sets, silence periods, and wave files.

Attribute	Permitted Values	Default	Interpretation
se.stutterOnVoiceMail	0, 1	1	If set to 1, stuttered dial tone is used in place of normal dial tone to indicate that one or more messages (voice-mail) are waiting at the message center.
se.appLocalEnabled	0, 1	1	If set to 1, local user interface sound effects such as confirmation/error tones, will be enabled.

4.6.1.7.1 Patterns <patterns/>

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:

Instruction	Meaning	Example
sampled (n)	Play sampled audio file n[a]	se.pat.callProg.x.inst.y.type =”sampled” (sampled audio file instruction type) se.pat.callProg.x.inst.y.value =”3” (specifies sampled audio file 3)
chord (n, d)	Play chord set n (d is optional and allows the chord set ON duration to be overridden to d milliseconds)	se.pat.callProg.x.inst.y.type = “chord” (chord set instruction type) se.pat.callProg.x.inst.y.value = “3” (specifies call progress chord set 3) se.pat.callProg.x.inst.y.param = “2000” (override ON duration of chord set to 2000 milliseconds)
silence (d)	Play silence for d milliseconds (Rx audio is not muted)	se.pat.callProg.x.inst.y.type = “silence” (silence instruction type) se.pat.callProg.x.inst.y.value = “300” (specifies silence is to last 300 milliseconds)

Instruction	Meaning	Example
branch (n)	Advance n instructions and execute that instruction (n must be negative and must not branch beyond the first instruction)	se.pat.callProg.x.inst.y.type = “branch” (branch instruction type) se.pat.callProg.x.inst.y.value = “-5” (step back 5 instructions and execute that instruction)

a. Currently, patterns that use the *sampled* instruction are limited to the following format: *sampled* followed by optional *silence* and optional *branch* back to the beginning.

In the following table, *x* is the pattern number, *y* is the instruction number. Both *x* and *y* need to be sequential. There are three categories of sound effect patterns: *callProg* (call progress patterns), *ringer* and *misc* (miscellaneous).

Attribute	Permitted Values	Interpretation
se.pat.callProg.x.name	UTF-8 encoded string	Used for identification purposes in the user interface (currently used for ringer patterns only); for patterns that use a sampled audio file which has been overridden by a downloaded replacement, the se.pat.ringer.x.name parameter will be overridden in the user interface by the file names of the wave file.
se.pat.callProg.x.inst.y.type	sampled OR chord OR silence OR branch	As above.
se.pat.callProg.x.inst.y.value	integer	Instruction type: — Interpretation: sampled — sampled audio file number chord — chord set number silence — silence duration in ms branch — number of instructions to advance
se.pat.callProg.x.inst.y.param	positive integer	If instruction type is chord, this optional parameter specifies the on duration to be used, overriding the on duration specified in the chord-set definition.

4.6.1.7.1.1 Call Progress Patterns

The following table maps call progress patterns to their usage within the phone.

Call progress pattern number	Use within phone
1	dial tone
2	busy tone
3	ring back tone
4	reorder tone
5	stuttered dial tone
6	call waiting tone
7	alternate call waiting tone (distinctive)
8	confirmation tone
9	howler tone (off-hook warning)
10	record warning
11	message waiting tone
12	alerting
13	intercom announcement tone
14	barge-in tone

4.6.1.7.1.2 Ringer Patterns

The following table maps ringer pattern numbers to their default descriptions.

Ringer pattern number	Default description
1	Silent Ring[a]
2	Low Trill
3	Low Double Trill
4	Medium Trill
5	Medium Double Trill
6	High Trill
7	High Double Trill
8	Highest Trill

Ringer pattern number	Default description
9	Highest Double Trill
10	Beeble
11	Triplet
12	Ringback-style
13	Sampled audio file 2[b]
14	Sampled audio file 3
15	Sampled audio file 4
16	Sampled audio file 5
17	Sampled audio file 6
18	Sampled audio file 7
19	Sampled audio file 8
20	Sampled audio file 9
21	Sampled audio file 10
22	Sampled audio file 11

a. Silent Ring will only provide a visual indication of an incoming call, but no audio indication.
b. Sampled audio files 1-21 all use the same built-in file unless that file has been replaced with a downloaded file. For more information, see 4.6.1.6 Sampled Audio for Sound Effects <sampled_audio/> on page 85.

4.6.1.7.1.3 Miscellaneous Patterns

The following table maps miscellaneous patterns to their usage within the phone.

Miscellaneous pattern number	Use within phone
1	new message waiting indication
2	new instant message
3	Not used.
4	local hold notification
5	positive confirmation
6	negative confirmation
7	welcome (boot up)

4.6.1.7.2 Ring type <ringType/>

Ring type 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 attributes 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 indication) of incoming call (no ringer needs to be specified).
answer	Provide auto-answer on incoming call[a].
ring-answer	Provide auto answer on incoming call after a ring period[a].

a. Note that auto-answer on incoming call is currently only applied if there is no other call in progress on the phone at the time.

In the following table, *x* is the ring class number. The *x* index needs to be sequential.

Attribute	Permitted Values	Interpretation
se.rt.enabled	0,1	Set to 1 to enable the ring type feature within the phone, 0 otherwise.
se.rt.modification.enabled	0,1	Set to 1 to allow user modification via local user interface of the pre-defined ring type enabled for modification[a].
se.rt.x.name	UTF-8 encoded string	Used for identification purposes in the user interface[a].
se.rt.x.type	ring OR visual OR answer OR ring-answer	As defined in table above.
se.rt.x.ringer	integer - only relevant if the type is set to ‘ring’ or ‘ring-answer’	The ringer index to be used for this class of ring. The ringer index should match one of 4.6.1.7.1.2 Ringer Patterns on page 89.
se.rt.x.callWait	integer - only relevant if the type is set to ‘ring’ or ‘ring-answer’	The call waiting index to be used for this class of ring. The call waiting index should match one defined in 4.6.1.7.1.1 Call Progress Patterns on page 89.
se.rt.x.timeout	positive integer - only relevant if the type is set to ‘ring-answer’. Default value is 2000.	The duration of the ring in milliseconds before the call is auto answered. If this field is omitted or is left blank, a value of 2000 is used.

Attribute	Permitted Values	Interpretation
se.rt.x.mod	0,1	Set to 1 if the user interface should allow for modification by the user of the ringer index used for this ring class.

a. Modification via user interface will be implemented in a future release.

4.6.1.8 Voice Settings <voice/>

4.6.1.8.1 Voice Coding Algorithms <codecs/>

The following voice codecs are supported:

Algorithm	MIME Type	Label	Bit Rate	Sample Rate	Frame Size	Effective Audio Bandwidth
G.711μ-law	PMCU	G711mu	64 Kbps	8 Ksps	10ms - 80ms	3.5KHz
G.711a-law	PCMA	G711A	64 Kbps	8 Ksps	10ms - 80ms	3.5KHz
G.729AB	G729	G729AB	8 Kbps	8 Ksps	10ms - 80ms	3.5KHz

4.6.1.8.1.1 Codec Preferences <preferences/>

Attribute	Permitted Values	Default	Interpretation
voice.codecPref.G711Mu	Null, 1-3	1	Specifies the codec preferences for SoundPoint® IP 500, 501, 600 and 601 platforms. 1 = highest 3 = lowest Null = do not use Give each codec a unique priority, this will dictate the order used in SDP negotiations.
voice.codecPref.G711A		2	
voice.codecPref.G729AB		3	

Attribute	Permitted Values	Default	Interpretation
voice.codecPref.IP_300.G711Mu	Null, 1-3	1	Specifies the codec preferences for SoundPoint® IP 300 and 301 platforms. 1 = highest 3 = lowest Null = do not use Give each codec a unique priority, this will dictate the order used in SDP negotiations.
voice.codecPref.IP_300.G711A		2	
voice.codecPref.IP_300.G729AB		3	
voice.codecPref.IP_4000.G711Mu	Null, 1-3	1	Specifies the codec preferences for the SoundStation® IP 4000 platform. Interpretation as above.
voice.codecPref.IP_4000.G711A		2	
voice.codecPref.IP_4000.G729AB		Null	Not supported by default so that G.711Mu and G.711A local conferences can be supported. This restriction will be removed in a future release.

4.6.1.8.1.2 Codec Profiles <profiles/>

The following profile attributes can be adjusted for each of the three supported codecs. In the table, *x*=G711Mu, G711A, or G729AB.

Attribute	Permitted Values	Interpretation
voice.audioProfile.x.payloadSize	10, 20, 30, ...80	Preferred Tx payload size in milliseconds to be provided in SDP offers and used in the absence of ptime negotiations. This is also the range of supported Rx payload sizes.
voice.audioProfile.x.jitterBufferMin	40, 50, 60, ... (multiple of 10)	The smallest jitter buffer depth (in milliseconds) that must be achieved before play out begins for the first time. Once this depth has been achieved initially, the depth may fall below this point and play out will still continue. This parameter should be set to the smallest possible value which is at least two packet payloads, and larger than the expected short term average jitter.

Attribute	Permitted Values	Interpretation
voice.audioProfile.x.jitterBufferShrink	10, 20, 30, ... (multiple of 10)	The absolute minimum duration time (in milliseconds) of RTP packet Rx with no packet loss between jitter buffer size shrinks. Use smaller values (1000 ms) to minimize the delay on known good networks. Use larger values to minimize packet loss on networks with large jitter (3000 ms).
voice.audioProfile.x.jitterBufferMax	> jitterBufferMin, multiple of 10, <=500 for IP 500, 501 and 600, <= 160 for IP 300 and 301	The largest jitter buffer depth to be supported (in milliseconds). Jitter above this size will always cause lost packets. This parameter should be set to the smallest possible value that will support the expected network jitter.

4.6.1.8.2 Volume Persistence <volume/>

The user's selection of the receive volume during a call can be remembered between calls. This can be configured per termination (handset, headset and hands-free/chassis). In some countries regulations exist which dictate that receive volume should be reset to nominal at the start of each call on handset and headset.

Attribute	Permitted Values	Default	Interpretation
voice.volume.persist.handset	0, 1	0	If set to 1, the receive volume will be remembered between calls. If set to 0, the receive volume will be reset to nominal at the start of each call.
voice.volume.persist.headset	0, 1	0	
voice.volume.persist.handsfree	0, 1	1	

4.6.1.8.3 Gains <gains/>

The default gain settings have been carefully adjusted to comply with the TIA-810-A digital telephony standard.

Do not alter these values.

Attribute	Default
voice.gain.rx.analog.handset	0
voice.gain.rx.analog.headset	0
voice.gain.rx.analog.chassis	-3
voice.gain.rx.analog.chassis.IP_300	-6
voice.gain.rx.analog.chassis.IP_4000	3
voice.gain.rx.analog.chassis.IP_601	0
voice.gain.rx.analog.ringer	-3
voice.gain.rx.analog.ringer.IP_300	-6
voice.gain.rx.analog.ringer.IP_4000	3
voice.gain.rx.analog.ringer.IP_601	0
voice.gain.rx.digital.handset	-15
voice.gain.rx.digital.headset	-21
voice.gain.rx.digital.chassis	0
voice.gain.rx.digital.chassis.IP_4000	0
voice.gain.rx.digital.chassis.IP_601	0
voice.gain.rx.digital.ringer	-21
voice.gain.rx.digital.ringer.IP_4000	-21
voice.gain.rx.digital.ringer.IP_601	-21
voice.gain.rx.analog.handset.sidetone	-14
voice.gain.rx.analog.headset.sidetone	-24
voice.gain.tx.analog.handset	12
voice.gain.tx.analog.headset	3
voice.gain.tx.analog.chassis	6
voice.gain.tx.analog.chassis.IP_300	0
voice.gain.tx.analog.chassis.IP_4000	3
voice.gain.tx.analog.chassis.IP_601	6
voice.gain.tx.digital.handset	0

Attribute	Default
voice.gain.tx.digital.headset	0
voice.gain.tx.digital.chassis	3
voice.gain.tx.digital.chassis.IP_4000	0
voice.gain.tx.digital.chassis.IP_601	3
voice.gain.tx.analog.preamp.handset	14
voice.gain.tx.analog.preamp.headset	23
voice.gain.tx.analog.preamp.chassis	32
voice.gain.tx.analog.preamp.chassis.IP_601	32

4.6.1.8.4 Acoustic Echo Cancellation <AEC/>

These settings control the performance of the speakerphone acoustic echo canceller.

Do not alter these values.

Attribute	Default
voice.aec.hs.enable	0
voice.aec.hs.lowFreqCutOff	100
voice.aec.hs.highFreqCutOff	7000
voice.aec.hs.erlTab_0_300	-24
voice.aec.hs.erlTab_300_600	-24
voice.aec.hs.erlTab_600_1500	-24
voice.aec.hs.erlTab_1500_3500	-24
voice.aec.hs.erlTab_3500_7000	-24
voice.aec.hd.enable	0
voice.aec.hd.lowFreqCutOff	100
voice.aec.hd.highFreqCutOff	7000
voice.aec.hd.erlTab_0_300	-24
voice.aec.hd.erlTab_300_600	-24
voice.aec.hd.erlTab_600_1500	-24
voice.aec.hd.erlTab_1500_3500	-24
voice.aec.hd.erlTab_3500_7000	-24

Attribute	Default
voice.aec.hf.enable	1
voice.aec.hf.lowFreqCutOff	100
voice.aec.hf.highFreqCutOff	7000
voice.aec.hf.erlTab_0_300	-6
voice.aec.hf.erlTab_300_600	-6
voice.aec.hf.erlTab_600_1500	-6
voice.aec.hf.erlTab_1500_3500	-6
voice.aec.hf.erlTab_3500_7000	-6

4.6.1.8.5 Acoustic Echo Suppression <AES/>

These settings control the performance of the speakerphone acoustic echo suppressor.

Do not alter these values.

Attribute	Default
voice.aes.hs.enable	0
voice.aes.hs.duplexBalance	7
voice.aes.hd.enable	0
voice.aes.hd.duplexBalance	0
voice.aes.hf.enable	1
voice.aes.hf.duplexBalance.0	7
voice.aes.hf.duplexBalance.1	7
voice.aes.hf.duplexBalance.2	6
voice.aes.hf.duplexBalance.3	6
voice.aes.hf.duplexBalance.4	5
voice.aes.hf.duplexBalance.5	4
voice.aes.hf.duplexBalance.6	4
voice.aes.hf.duplexBalance.7	3
voice.aes.hf.duplexBalance.8	2
voice.aes.hf.duplexBalance.IP_4000.0	10
voice.aes.hf.duplexBalance.IP_4000.1	9
voice.aes.hf.duplexBalance.IP_4000.2	8
voice.aes.hf.duplexBalance.IP_4000.3	7
voice.aes.hf.duplexBalance.IP_4000.4	6
voice.aes.hf.duplexBalance.IP_4000.5	5
voice.aes.hf.duplexBalance.IP_4000.6	4
voice.aes.hf.duplexBalance.IP_4000.7	3
voice.aes.hf.duplexBalance.IP_4000.8	2

4.6.1.8.6 Background Noise Suppression <NS/>

These settings control the performance of the transmit background noise suppression feature.

Do not alter these values.

Attribute	Default
voice.ns.hs.enable	0
voice.ns.hs.signalAttn	-6
voice.ns.hs.silenceAttn	-9
voice.ns.hd.enable	0
voice.ns.hd.signalAttn	0
voice.ns.hd.silenceAttn	0
voice.ns.hf.enable	1
voice.ns.hf.signalAttn	-6
voice.ns.hf.silenceAttn	-9
voice.ns.hf.IP_4000.enable	1
voice.ns.hf.IP_4000.signalAttn	-6
voice.ns.hf.IP_4000.silenceAttn	-9

4.6.1.8.7 Automatic Gain Control <AGC/>

These settings control the performance of the transmit automatic gain control feature.[9]

Do not alter these values.

Attribute	Default
voice.agc.hs.enable	0
voice.agc.hd.enable	0
voice.agc.hf.enable	0

9. Automatic Gain Control will be implemented in a future release.

4.6.1.8.8 Receive Equalization <RXEQ/>

These settings control the performance of the receive equalizaton feature.

Do not alter these values.

Attribute	Default
voice.rxEq.hs.IP_500.preFilter.enable	1
voice.rxEq.hs.IP_600.preFilter.enable	1
voice.rxEq.hs.IP_601.preFilter.enable	1
voice.rxEq.hs.IP_500.postFilter.enable	0
voice.rxEq.hs.IP_600.postFilter.enable	0
voice.rxEq.hs.IP_601.postFilter.enable	0
voice.rxEq.hd.IP_500.preFilter.enable	0
voice.rxEq.hd.IP_600.preFilter.enable	0
voice.rxEq.hd.IP_601.preFilter.enable	0
voice.rxEq.hd.IP_500.postFilter.enable	0
voice.rxEq.hd.IP_600.postFilter.enable	0
voice.rxEq.hd.IP_601.postFilter.enable	0
voice.rxEq.hf.IP_500.preFilter.enable	1
voice.rxEq.hf.IP_600.preFilter.enable	1
voice.rxEq.hf.IP_601.preFilter.enable	1
voice.rxEq.hf.IP_4000.preFilter.enable	0
voice.rxEq.hf.IP_500.postFilter.enable	1
voice.rxEq.hf.IP_600.postFilter.enable	1
voice.rxEq.hf.IP_601.postFilter.enable	1
voice.rxEq.hf.IP_4000.postFilter.enable	0
voice.rxEq.hf.IP_500.preFilter...	Do not change these values.

4.6.1.8.9 Transmit Equalization <TXEQ/>

These settings control the performance of the hands-free transmit equalization feature.

Do not alter these values.

Attribute	Default
voice.txEq.hs.IP_500.preFilter.enable	0
voice.txEq.hs.IP_600.preFilter.enable	0
voice.txEq.hs.IP_601.preFilter.enable	0
voice.txEq.hs.IP_500.postFilter.enable	1
voice.txEq.hs.IP_600.postFilter.enable	1
voice.txEq.hs.IP_601.postFilter.enable	1
voice.txEq.hd.IP_500.preFilter.enable	0
voice.txEq.hd.IP_600.preFilter.enable	0
voice.txEq.hd.IP_601.preFilter.enable	0
voice.txEq.hd.IP_500.postFilter.enable	0
voice.txEq.hd.IP_600.postFilter.enable	0
voice.txEq.hd.IP_601.postFilter.enable	0
voice.txEq.hf.IP_500.preFilter.enable	0
voice.txEq.hf.IP_600.preFilter.enable	0
voice.txEq.hf.IP_601.preFilter.enable	0
voice.txEq.hf.IP_4000.preFilter.enable	0
voice.txEq.hf.IP_500.postFilter.enable	1
voice.txEq.hf.IP_600.postFilter.enable	1
voice.txEq.hf.IP_601.postFilter.enable	1
voice.txEq.hf.IP_4000.postFilter.enable	0
voice.txEq.hf.IP_500.postFilter...	Do not change these values.

4.6.1.8.10 Voice Activity Detection <VAD/>

These settings control the performance of the voice activity detection (silence suppression) feature.

Attribute	Permitted Values	Default	Interpretation
voice.vadEnable	0, 1	0	If set to 1, enable VAD.
voice.vadThresh	integer from 0 to 30	15	The threshold for determining what is active voice and what is background noise in dB. This does not apply to G.729AB codec operation which has its own built-in VAD function.

4.6.1.9 Quality of Service <QOS/>

These settings control the Quality of Service (QOS) options.

4.6.1.9.1 Ethernet IEEE 802.1p/Q <Ethernet/>

These settings control the 802.1p/Q user_priority field.

4.6.1.9.1.1 RTP <RTP/>

These parameters apply to RTP packets.

Attribute	Permitted Values	Default	Interpretation
qos.ethernet.rtp.user_priority	0-7	5	User-priority used for RTP packets.

4.6.1.9.1.2 Call Control <CallControl/>

These parameters apply to call control packets, such as the network protocol signaling.

Attribute	Permitted Values	Default	Interpretation
qos.ethernet.callControl.user_priority	0-7	5	User-priority used for call control packets.

4.6.1.9.1.3 Other <Other/>

These default parameter values are used for all packets which are not set explicitly.

Attribute	Permitted Values	Default	Interpretation
qos.ethernet.other.user_priority	0-7	2	User-priority used for packets that do not have a per-protocol setting.

4.6.1.9.2 IP TOS <IP/>

These settings control the "type of service" field in outgoing packets.

4.6.1.9.2.1 RTP <RTP/>

These parameters apply to RTP packets.

Attribute	Permitted Values	Default	Interpretation
qos.ip.rtp.min_delay	0, 1	1	If set to 1, set min-delay bit in the IP TOS field of the IP header, or else don't set it.
qos.ip.rtp.max_throughput	0, 1	1	If set to 1, set max-throughput bit in the IP TOS field of the IP header, or else don't set it.
qos.ip.rtp.max_reliability	0, 1	0	If set to 1, set max-reliability bit in the IP TOS field of the IP header, or else don't set it.
qos.ip.rtp.min_cost	0, 1	0	If set to 1, set min-cost bit in the IP TOS field of the IP header, or else don't set it.
qos.ip.rtp.precedence	0-7	5	If set to 1, set precedence bits in the IP TOS field of the IP header, or else don't set them.

4.6.1.9.2.2 Call Control <CallControl/>

These parameters apply to call control packets, such as the network protocol signaling.

Attribute	Permitted Values	Default	Interpretation
qos.ip.callControl.min_delay	0, 1	1	If set to 1, set min-delay bit in the IP TOS field of the IP header, or else don't set it.
qos.ip.callControl.max_throughput	0, 1	0	If set to 1, set max-throughput bit in the IP TOS field of the IP header, or else don't set it.
qos.ip.callControl.max_reliability	0, 1	0	If set to 1, set max-reliability bit in the IP TOS field of the IP header, or else don't set it.
qos.ip.callControl.min_cost	0, 1	0	If set to 1, set min-cost bit in the IP TOS field of the IP header, or else don't set it.
qos.ip.callControl.precedence	0-7	5	If set to 1, set precedence bits in the IP TOS field of the IP header, or else don't set them.

4.6.1.10 Basic TCP/IP <TCP_IP/>

4.6.1.10.1 Network Monitoring <netMon/>

Do not alter these values.

Attribute	Permitted Values	Default
tcpIpApp.netMon.enabled	0, 1	1
tcpIpApp.netMon.period	1 to 86400	30

4.6.1.10.2 Time Synchronization <SNTP/>

The following table describes the parameters used to set up time synchronization and daylight savings time. The defaults shown will enable daylight savings time for North America.

Daylight savings defaults:

- do not use fixed day, use first or last day of week in the month,
- start DST on the first Sunday in April at 2 am,

- stop DST on the last Sunday in October at 2 am.

Attribute	Permitted Values	Default	Interpretation
tcpIpApp.sntp.resyncPeriod	positive integer	86400 (24 hours)	Time in seconds between SNTP re-syncs.
tcpIpApp.sntp.address[a]	valid host name or IP address	clock	Address of the SNTP server.
tcpIpApp.sntp.gmtOffset	positive or negative integer	-28800 (Pacific time)	Offset in seconds of the local time zone from GMT. Note: 3600 seconds per hour
tcpIpApp.sntp.daylightSavings.enable	0, 1	1	If set to 1, apply day-light savings rules to displayed time.
tcpIpApp.sntp.daylightSavings.fixedDay-Enable	0, 1	0	If set to 1, “April 1st” is used, otherwise “the first Sunday in April” is used.
tcpIpApp.sntp.daylightSavings.start.month	1-12	4 (April)	Month to start DST. 1=Jan, 2=Feb, ..., 12=Dec
tcpIpApp.sntp.daylightSavings.start.date	1-31	1	Day of the month to start DST.
tcpIpApp.sntp.daylightSavings.start.time	0-23	2	Time of day to start DST, in 24 hour clock. 2=2 am, 14=2 pm
tcpIpApp.sntp.daylightSavings.start.dayOf-Week	1-7	1	Day of week to apply DST. 1=Sun, 2=Mon, ..., 7=Sat
tcpIpApp.sntp.daylightSavings.start.dayOf-Week.lastInMonth	0, 10	0	If set to 1 and fixedDay-Enable=0, start DST on the last day of the week (specified by dayOf-Week) in the month, rather than the first in the month.
tcpIpApp.sntp.daylightSavings.stop.month	1-12	10	Month to stop DST. 1=Jan, 2=Feb, ..., 12=Dec

Attribute	Permitted Values	Default	Interpretation
tcpIpApp.sntp.daylightSavings.stop.date	1-31	1	Day of the month to start DST.
tcpIpApp.sntp.daylightSavings.stop.time	0-23	2	Time of day to stop DST, in 24 hour clock. 2= 2 am, 14=2 pm
tcpIpApp.sntp.daylightSavings.stop.dayOf-Week	1-7	1	Day of week to stop DST. 1=Sun, 2=Mon, ..., 7=Sat
tcpIpApp.sntp.daylightSavings.stop.dayOf-Week.lastInMonth	0, 1	1	If set to 1 and fixedDay-Enable=0, stop DST on the last day of the week (specified by dayOf-Week) in the month, rather than the first in the month.

a. Both tcpIpApp.sntp.address and tcpIpApp.sntp.gmtOffset can be provided via DHCP. If so, the DHCP parameters will override the parameters in sip.cfg.

4.6.1.10.3 port <port/>

4.6.1.10.3.1 RTP <RTP/>

Attribute	Permitted Values	Default	Interpretation
tcpIpApp.port.rtp.filterByIp	0, 1	1	If set to 1, reject RTP packets arriving from (sent from) a non-negotiated (via SDP) IP address.
tcpIpApp.port.rtp.filterByPort	0, 1	0	If set to 1, reject RTP packets arriving from (sent from) a non-negotiated (via SDP) port.

Attribute	Permitted Values	Default	Interpretation
tcpIpApp.port.rtp.forceSend	Null, 1024-65534	Null	When non-Null, send all RTP packets to, and expect all RTP packets to arrive on, the specified port. Note: both tcpIpApp.port.rtp.filterByIp and tcpIpApp.port.rtp.filterByPort must be enabled for this to work.
tcpIpApp.port.rtp.mediaPortRangeStart	Null, even integer from 1024-65534	Null	If set to Null, the value 2222 will be used for the first allocated RTP port, otherwise, the specified port will be used. Subsequent ports will be allocated from a pool starting with the specified port plus two up to a value of (start-port + 46), after which the port number will wrap back to the starting value.

4.6.1.11 Web Server <HTTPD/>

The phone contains a local web 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.

Attribute	Permitted Values	Default	Interpretation
httpd.enabled	0, 1	1	If set to 1, the HTTP server will be enabled.

4.6.1.11.1 Configuration <cfg/>

Attribute	Permitted Values	Default	Interpretation
httpd.cfg.enabled	0, 1	1	If set to 1, the HTTP server configuration interface will be enabled.

Attribute	Permitted Values	Default	Interpretation
httpd.cfg.port	1-65535	80	Port is 80 for HTTP servers. Care should be taken when choosing an alternate port.

4.6.1.12 Call Handling Configuration <call/>

Attribute	Permitted Values	Default	Interpretation
call.rejectBusyOnDnd	0, 1	1	If set to 1, reject all incoming calls with the reason “busy” if do-not-disturb is enabled.
call.enableOnNotRegistered	0, 1	1	If set to 1, calls will be allowed when the phone is not successfully registered, otherwise, calls will not be permitted without a valid registration.
call.offeringTimeOut	positive integer	60	Time in seconds to allow an incoming call to ring before dropping the call, 0=infinite[a].
call.ringBackTimeOut	positive integer	60	Time in seconds to allow an outgoing call to remain in the ringback state before dropping the call, 0=infinite.
call.lastCallReturnString	string of maximum length 32	*69	The string sent to the server when the user selects the “last call return” action.
call.callWaiting.prompt	0, 1	0	If set to 1, an incoming call received when another call is active will change the User Interface focus (call appearance and soft keys).
call.callsPerLineKey	1 to 24 OR 1 to 8	24 OR 8	For the SoundPoint® IP 600 and 601 the permitted range is 1 to 24 and the default is 24. For all other phones the permitted range is 1 to 8 and the default is 8. This is the number of calls or conferences which may be active or on hold per line key on the phone. Note that this may be overridden by the per-registration attribute of reg.x.callsPerLineKey. See 4.6.2.1 Registration <reg/> on page 128.

a. The call diversion, no answer feature will take precedence over this feature if enabled. For more information, see 4.6.2.3.3 No Answer <noanswer/> on page 134.

4.6.1.12.1 Shared Calls <shared/>

Attribute	Permitted Values	Default	Interpretation
call.shared.disableDivert	0, 1	1	If set to 1, disable diversion feature for shared lines.
call.shared.seizeFailReorder	0, 1	1	If set to 1, play re-order tone locally on shared line seize failure.
call.shared.oneTouchResume	0, 1	0	Note: This parameter affects the Sound-Station® IP 4000 phone only. For other phones a quick press and release of the line key will resume a call whereas press-ing and holding down the line key will show a list of calls on that line. If set to 1, when a shared line has a call on hold the remote user can press that line and resume the call. If more than one call is on hold on the line then the first one will be selected and resumed automatically. If set to 0, pressing the shared line will bring up a list of the calls on that line and the user can select which call the next action should be applied to.

4.6.1.12.2 Hold, Local Reminder <hold/><localReminder/>

Attribute	Permitted Values	Default	Interpretation
call.hold.localReminder.enabled	0, 1	0	If set to 1, periodically notify the local user that calls have been on hold for an extended period of time.
call.hold.localReminder.period	non-negative integer	60	Time in seconds between subse-quent reminders.
call.hold.localReminder.startDelay	non-negative integer	90	Time in seconds to wait before the initial reminder.

4.6.1.13 Directory <directory/>

The directory is stored in either flash memory or RAM on the phone. The directory size is limited based on the amount of flash memory in the phone[10].

When the volatile storage option is enabled, ensure that a properly configured boot server that allows uploads is available to store a back-up copy of the directory or its contents will be lost when the phone reboots or loses power.

Attribute	Permitted Values	Default	Interpretation
dir.local.volatile.2meg	0, 1	0	Attribute applies to platforms with 2 Mbytes of flash memory. If set to 1, use volatile storage for phone-resident copy of the directory to allow for larger size.
dir.local.nonVolatile.maxSize.2meg	1 to 20	20	Attribute applies to platforms with 2 Mbytes of flash memory. Maximum size in Kbytes of non-volatile storage that the directory will be permitted to consume.
dir.local.volatile.4meg	0, 1	0	Applies to platforms with 4 Mbytes of flash memory. If set to 1, use volatile storage for phone-resident copy of the directory to allow for larger size.
dir.local.nonVolatile.maxSize.4meg	1 to 50	50	Applies to platforms with 4 Mbytes of flash memory. Maximum size in Kbytes of non-volatile storage that the directory will be permitted to consume.
dir.local.volatile.maxSize	1 to 100	100	Maximum size in Kbytes of volatile storage that the directory will be permitted to consume.

10.The phone could have 2 megabytes or 4 megabytes of flash memory depending on the hardware model.

4.6.1.14 Fonts <font/>

This section does not apply to the SoundPoint® IP 300 and 301 phones.

These settings control the phone's ability to dynamically load an external font file during boot up. Loaded fonts can either overwrite pre-existing fonts embedded within the software (not recommended) or can extend the phone's font support for Unicode ranges not already embedded. The font file must be a Microsoft .fnt or .fon[11] file format. The font file name must follow a specific pattern as described:

- Font file name: <fontName>_<fontHeightInPixels>_<fontRange>.<fontExtension>
- <fontName> is a free string of characters that typically carries the meaning of the font. Examples are "fontFixedSize" for a fixed-size font, or "fontProportionalSize" for a proportional size font.
- <fontHeightInPixels> describes the font height in number of screen pixels.
- <fontRange> describes the Unicode range covered by this font. Since .fnt or .fon are 256 characters based blocks, the <fontRange> is Uxx00_UxxFF (.fnt file) or Uxx00_UyyFF (.fon file). For more information, see 3.5.1 Multilingual User Interface on page 52.
- <fontExtension> describes the file type. Either .fnt for single 256 characters font or .fon for multiple .fnt files.

If it is necessary to overwrite an existing font, use these <fontName>_<fontHeightInPixels>:

SoundPoint® IP 500 and 501	
"fontProp_10"	This is the font used widely in the current implementation.
"fontPropSoftkey_10"	This is the soft key specific font.
SoundPoint® IP 600 and 601	
"fontProp_19"	This is the font used widely in the current implementation including for soft keys.
"fontProp_26"	This is the font used to display time (but not date).
"fontProp_x"	This is a small font used for the CPU/Load/Net utilization graphs, this is the same as the "fontProp_10" for the SoundPoint® IP 500.

If the <fontName>_<fontHeightInPixels> does not match any of the names above, then the downloaded font will be applied against all fonts defined in the phone, which means that you may lose the benefit of fonts being calibrated differently depending on their usage. For example, the font used to display the time on the Sound Point® IP 600 is a large font, larger than the one used to display the date, and if you overwrite this default font with a unique font, you lose this size aspect.

11..fon file format is a collection of .fnt fonts mangled together within a single file.

Example of use:

- to overwrite the font used for SoundPoint® IP 500 soft keys for ASCII, the name should be “fontPropSoftkey_10_U0000_U00FF.fnt”
- to add support for a new font that will be used everywhere and that is not currently supported. For example, for the Eastern/Central European Czech language, this is Unicode range 100-17F, the name could be “fontCzechIP500_10_U0100_U01FF.fnt” and “fontCzechIP600_19_U0100_U01FF.fnt”

When defining a single .fon file, there is a need for a “font delimiter”, currently “Copyright Polycom Canada Ltd” is used as an embedded delimiter, but this can be configured using “font.delimiter”. The font delimiter is important to retrieve the different mangled .fnt blocks. This font delimiter must be placed in the “copyright” attribute of the .fnt header. .fon files are useful if you want to include support for a large number of font ranges at once, otherwise, if simply adding or changing a few fonts currently in use, multiple .fnt files are recommended since they are easier to work with individually.

Attribute	Permitted Values	Default	Interpretation
font.delimiter	string up to 256 ASCII characters	Null	Delimiter required to retrieve different mangled .fnt blocks.

4.6.1.14.1 IP_500 font <IP_500/>

Attribute	Permitted Values	Default	Interpretation
font.IP_500.x.name	fontName_height_Uxx00_UyyFF.fon OR fontName_height_Uxx00_UxxFF.fnt	Null	Defines the font file that will be loaded from boot server during boot up. Note: When several font.IP_500.x.names are defined, the index x must follow consecutive increasing order.

4.6.1.14.2 IP_600 font <IP_600/>

Attribute	Permitted Values	Default	Interpretation
font.IP_600.x.name	fontName_height_Uxx00_UyyFF.fon OR fontName_height_Uxx00_UxxFF.fnt	Null	Defines the font file that will be loaded from boot server during boot up. Note: When several font.IP_600.x.names are defined, the index x must follow consecutive increasing order.

4.6.1.15 Keys <keys/>

These settings control the scrolling behavior of keys and can be used to change key functions.

Attribute	Permitted Values	Default	Interpretation
key.scrolling.timeout	positive integer	1	The time-out after which a key that is enabled for scrolling will go into scrolling mode until the key is released. Keys enabled for scrolling are menu navigation keys (left, right, up, down arrows), volume keys, and some context-specific soft keys. The value is an integer multiple of 500 milliseconds (1=500ms).

SoundPoint® IP 300, 301, 500, 501 and 600 key functions can be changed from the factory defaults, although this is typically not necessary. For each key whose function you wish to change, add an XML attribute in the format described in the following table to the <keys .../> element of the configuration file. These will override the built-in assignments.

Remapping the arrow keys is not recommended.

In the following table, *x*=IP_300, IP_500 or IP_600, *y* is the key number. Note that IP_300 parameters affect SoundPoint® IP 300 and 301 phones, and IP_500 parameters affect SoundPoint® IP 500 and 501 phones. IP 300: *y*=1-35; IP 500: *y*=1-40; IP 600: *y*=1-42

Attribute	Permitted Values	Interpretation
key.x.y.function.prim	Functions listed below.	Sets the function for key y on platform x.

Attribute	Permitted Values	Interpretation
key.x.y.subPoint.prim	positive integer	Sets the sub-identifier for key functions with a secondary array identifier such as SpeedDial.

The following table lists the functions that are available:

Function
ArrowDown
ArrowLeft
ArrowRight
ArrowUp
BuddyStatus
CallList
Conference
Delete
Dialpad0
Dialpad1
Dialpad2
Dialpad3
Dialpad4
Dialpad5
Dialpad6
Dialpad7
Dialpad8
Dialpad9
DialpadStar
DialpadPound
Directories
DoNotDisturb
Handsfree
Headset
Hold

Function
Line1
Line2
Line3
Line4
Line5
Line6
Messages
Menu
MicMute
MyStatus
Null
Offline
Redial
Select
Setup
SoftKey1
SoftKey2
SoftKey3
SoftKey4
SpeedDial
SpeedDialMenu
Transfer
VolDown
VolUp

4.6.1.16 Bitmaps <bitmaps/>

Bitmaps used by the phone are defined in this section.

4.6.1.16.1 Platform <IP_300/>, <IP_500/>, <IP_600/> and <IP_4000/>

In the following table, *x*=IP_300, IP_500, IP_600, or IP_4000 and *y* is the bitmap number. Note that IP_300 parameters affect SoundPoint® IP 300 and 301 phones, IP_500 parameters affect SoundPoint® IP 500 and 501 phones and IP_600 parameters affect SoundPoint® IP 600 and 601 phones.

Attribute	Permitted Values	Interpretation
bitmap.x.y.name	The name of a bit-map to be used.	This is the name of a bitmap to be used for creating an animation. If the bitmap is to be downloaded from the boot server, its name must: 1. Be different from any name already in use in sip.cfg. 2. Match the name of the corresponding <file-Name>.bmp to be retrieved from the boot server.

4.6.1.17 Indicators <indicators/>

Indicators (graphic icons, animations, and LED patterns) used by the phone are defined in this section.

Attribute	Permitted Values	Default	Interpretation
ind.idleDisplay.enabled	0, 1	0	If set to 1, the idle display may support pre-sentation of a custom animation if config-ured properly in the animation section of sip.cfg.

4.6.1.17.1 Animations <Animations/> <IP_300/>, <IP_500/>, <IP_600/> and <IP_4000/>

This section defines bitmap animations composed of bitmap/duration couples. In the following table, *x*=IP_300, IP_500, IP_600 or IP_4000, *y* is the animation number, *z* is the step in the animation. Note that IP_300 parameters affect SoundPoint® IP 300 and

301 phones, IP_500 parameters affect SoundPoint® IP 500 and 501 phones and IP_600 parameters affect SoundPoint® IP 600 and 601 phones.

Attribute	Permitted Values	Interpretation
ind.anim.x.y.frame.z.bitmap	A bitmap name defined previously.	Bitmap to use. Note that it must be defined already, see 4.6.1.16.1 Platform <IP_300/>, <IP_500/>, <IP_600/> and <IP_4000/> on page 116.
ind.anim.x.y.frame.z.duration	positive integer	Duration in milliseconds for this step. 0=infinite.

4.6.1.17.2 Patterns <Patterns/>

This section defines patterns for the LED indicators. In the following table, *x* is the pattern number, *y* is the step in the pattern.

Attribute	Permitted Values	Interpretation
ind.pattern.x.step.y.state	On or Off	Turn LED on or off for this step.
ind.pattern.x.step.y.duration	positive integer	Duration in milliseconds for this step. 0=infinite
ind.pattern.x.step.y.colour	Red or Green (default is Red if not specified)	For bi-color LEDs, specify color.

4.6.1.17.3 Classes <Classes/>

This section defines the available classes for the LED and graphical icon indicator types. In the following table, *x* is the class number, *y* is the identifier of the state number for that class.

Attribute	Permitted Values	Interpretation
ind.class.x.state.y.index	positive integer	For LED type indicators, index refers to the pattern index, such as index x in the <Patterns/> tag above. For GraphicIcon type indicators, index refers to the animation index, such as index y in the <Animations/> tag above.

4.6.1.17.4 Assignments <Assignments/>

This section assigns a type, a class, and, in the case of the GraphicIcon type, a physical location and size in pixels on the LCD display or in the case of the LED type, a physical LED number.

4.6.1.17.4.1 LEDs <led/>

In the following table, *x* is the LED number.

Attribute	Permitted Values	Interpretation
ind.led.x.index		This is for internal usage only and should not be changed (this is the logical index).
ind.led.x.class	positive integer	Assigns the class (defined in 4.6.1.17.3 Classes <Classes/> on page 117) for this indicator.
ind.led.x.physNum		This maps the logical index to a specific physical LED.

4.6.1.17.4.2 Graphic Icons <gi/> <IP_300/>, <IP_500/>, <IP_600/> and <IP_4000/>

In the following table, *x*=IP_300, IP_500, IP_600 or IP_4000, *y* is the graphic icon number. Note that IP_300 parameters affect SoundPoint® IP 300 and 301 phones, IP_500 parameters affect SoundPoint® IP 500 and 501 phones and IP_600 parameters affect SoundPoint® IP 600 and 601 phones.

Attribute	Permitted Values	Interpretation
ind.gi.x.y.index		This is for internal usage only and should not be changed (this is the logical index).
ind.gi.x.y.class	positive integer	Assigns the class (defined in 4.6.1.17.3 Classes <Classes/> on page 117) for this indicator.
ind.gi.x.y.physX	IP 300: 0-19 IP 500: 0-159 IP 600: 0-319 IP 4000: 0-247	For GraphicIcon type indicators, this is the x-axis location of the upper left corner of the indictor measured in pixels from left to right.
ind.gi.x.y.physY	IP 300: 0-3 IP 500: 0-79 IP 600: 0-159 IP 4000: 0-67	For GraphicIcon type indicators, this is the y-axis location of the upper left corner of the indicator measured in pixels from top to bottom.

Attribute	Permitted Values	Interpretation
ind.gi.x.y.physW	IP 300: n/a IP 500: 1-160 IP 600: 1-320 IP 4000: 1-248	For GraphicIcon type indicators, this is the width of the indicator measured in pixels.
ind.gi.x.y.physH	IP 300: n/a IP 500: 1-80 IP 600: 1-160 IP 4000: 1-68	For GraphicIcon type indicators, this is the height of the indicator measured in pixels.

4.6.1.18 Event Logging <logging/>

Warning! Logging parameter changes can impair system operation. Do not change any logging parameters without prior consultation with Polycom.

The event logging system supports the following classes of events:

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

Example:

```
011511.006|so   |2|00|soCoreAudioTermChg: chassis -> idle
```

time stamp
ID
event class
missed events
text

Three formats are available for the event timestamp:

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

4.6.1.18.1 Basic Logging <level/><change/> and <render/>

Attribute	Permitted Values	Default	Interpretation
log.level.change.xxx	0-5	4	Control the logging detail level for individual components. These are the input filters into the internal memory-based log system.
log.render.level	0-6	1	Specifies the lowest class of event that will be rendered to the log files. This is the output filter from the internal memory-based log system.
log.render.type	0-2	2	See above table for timestamp type.
log.render.realtime	0, 1	1	Set to 1. Do not change.
log.render.stdout	0, 1	1	Set to 1. Do not change.
log.render.file	0, 1	1	Set to 1. Do not change.

Attribute	Permitted Values	Default	Interpretation
log.render.file.size	positive integer	16	Maximum local application log file size in Kbytes. When this size is exceeded, the file is uploaded to the boot server and the local copy is erased.
log.render.file.upload.period	positive integer	172800	Time in seconds between log file uploads to the boot server. Note: The log file will not be uploaded if no new events have been logged since the last upload.
log.render.file.upload.append	0, 1	1	If set to 1, use append mode when uploading log files to server. Note: HTTP and TFTP don't support append mode unless the server is set up for this.
log.render.file.upload.append.sizeLimit	positive integer	512	Maximum log file size on boot server in Kbytes.
log.render.file.upload.append.limit-Mode	delete, stop	delete	Behavior when server log file has reached its limit. delete=delete file and start over stop=stop appending to file

4.6.1.18.2 Scheduled Logging Parameters <scheduled/>

The phone can be configured to schedule certain advanced logging tasks on a periodic basis. These attributes should be set in consultation with Polycom. Each scheduled log task is controlled by a unique attribute set starting with log.sched.x where *x* identifies the task.

Attribute	Permitted Values	Interpretation
log.sched.x.name	alphanumeric string	Name of an internal system command to be periodically executed. To be supplied by Polycom.
log.sched.x.level	0-5	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 appear in the log.

Attribute	Permitted Values	Interpretation
log.sched.x.period	positive integer	Seconds between each command execution. 0=run once
log.sched.x.startMode	abs, rel	Start at *abs*olute time or *rel*ative to boot.
log.sched.x.startTime	positive integer OR hh:mm	Seconds since boot when startMode is *rel* or the start time in 24-hour clock format when startMode is *abs*.
log.sched.x.startDay	1-7	When startMode is *abs*, specifies the day of the week to start command execution. 1=Sun, 2=Mon, ..., 7=Sat

4.6.1.19 Security <security/>

These settings affect security aspects of the phone.

Attribute	Permitted Values	Default	Interpretation
sec.tagSerialNo	0, 1	0	If set to 1, the phone may advertise its serial number (Ethernet address) via protocol signaling.

4.6.1.19.1 Password Lengths <pwd/><length/>

Attribute	Permitted Values	Default	Interpretation
sec.pwd.length.admin	0-32	1	Password changes will need to be at least this long. Use 0 to allow null passwords.
sec.pwd.length.user	0-32	2	

4.6.1.20 Provisioning <provisioning/>

These settings control aspects of the phone's boot server provisioning system.

Attribute	Permitted Values	Default	Interpretation
prov.fileSystem.rfs0.minFreeSpace	5-512	5	**Note**: Changing these parameters is not advised. Minimum free space in Kbytes to reserve in the file system when downloading files from the boot server.
prov.fileSystem.ffs0.4meg.minFreeSpace		420	
prov.fileSystem.ffs0.2meg.minFreeSpace		48	
prov.polling.enabled	0, 1	0	If set to 1, automatic periodic boot server polling for upgrades is enabled.
prov.polling.mode	abs, rel	abs	Polling mode is *abs*olute or *rel*ative.
prov.polling.period	integer greater than 3600	86400	Polling period in seconds. Rounded up to the nearest number of days in *abs* mode. Measured relative to boot time in *rel* mode.
prov.polling.time	Format is hh:mm	03:00	Only used in *abs* mode. Polling time.

4.6.1.21 RAM Disk <RAMdisk/>

These settings control the phone's internal RAM disk feature. Changing these parameters is not advised.

Attribute	Permitted Values	Default	Interpretation
ramdisk.enable	0, 1	1	If set to 1, RAM disk will be available. The RAM disk is used to cache downloaded wave files, and other resources for the user interface.

Attribute	Permitted Values	Default	Interpretation
ramdisk.bytesPerBlock	0, 32, 33, ..., 1024	0	These three parameters use internal defaults when value is set to 0.
ramdisk.blocksPerTrack	0, 1, 2, ..., 65536	0	
ramdisk.nBlocks	0, 1, 2, ..., 65536	4096	
ramdisk.minsize	50 to 16384	50	Smallest size in Kbytes of RAM disk to create before returning an error. RAM disk size is variable depending on the amount of device memory.
ramdisk.minfree	512 to 16384	3072	Minimum amount of free space that must be left after the RAM disk has been created. The RAM disk's size will be reduced as necessary in order to leave this amount of free RAM.

4.6.1.22 Request <request/>

4.6.1.22.1 Delay <delay/>

These settings control the phone's behavior when a request for restart, reboot, or reconfiguration is received.

Attribute	Permitted Values	Default	Interpretation
request.delay.type	Null, "audio", or "call"	call	Defines the strategy to adopt before a request gets executed. If set to "audio", a request can be executed as soon as there is no active audio on the phone, independently of any call state. If set to "call", a request can be executed as soon as there are no calls in any state on the phone.

4.6.1.23 Feature <feature/>

These settings control the activation or deactivation of a feature at run time. In the table below, *x* is the feature number.

Attribute	Permitted Values	Interpretation
feature.x.name	"presence", "messaging", "directory", "calllist", "ring-download", "calllist-received", "calllist-placed", "calllist-missed", "url-dialing", "cpt-settings", "call-park", "group-call-pickup", "directed-call-pickup", "last-call-return", "acd-login-logout", "acd-agent-available"	These are features offered on the phone: • "presence" is the presence feature including management of buddies and own status • "messaging" is the instant messaging feature • "directory" is the local directory feature • "calllist" is the locally controlled call lists • "ring-download" is run-time downloading of ringers • "calllist-received" is the received-calls list feature (the "calllist" feature must be enabled for this feature to be available) • "calllist-placed" is the placed-calls list feature (the "calllist" feature must be enabled for this feature to be available) • "calllist-missed" is the missed-calls list feature (the "calllist" feature must be enabled for this feature to be available) • "url-dialing" controls whether URL/name dialing is available from a private line (it is never available from a shared line) • "cpt-settings" controls whether call progress tones can be selected by the phone user using the Settings menu • "call-park" is the call park and park-retrieve features • "group-call-pickup" is the group call pickup feature • "directed-call-pickup" is the directed call pickup feature • "last-call-return" is the last call return feature • "acd-login-logout" is the ACD login/logout feature • "acd-agent-available" is the ACD agent available/unavailable feature
feature.x.enabled	0 or 1 (default)	If set to 0, the feature will be disabled. If set to 1, the feature will be enabled and usable by the local user.

4.6.1.24 Resource <resource/>

These settings control the maximum size or an external resource retrieved at run time.

4.6.1.24.1 finder <finder/>

Attribute	Permitted Values	Default	Interpretation
res.finder.sizeLimit	positive integer	300	If a resource that is being downloaded to the phone is larger than this value * 1000 bytes (= the maximum size), the resource will be automatically truncated to the maximum size defined.

4.6.1.24.2 quotas <quotas/>

Attribute	Permitted Values	Interpretation
res.quotas.x.name	"tone", "bitmap", "font", or "xmlui"	The name of the sub-application for which the particular quota will apply: • "tone" relates to all downloaded tones and sound effects • "bitmap" relates to all downloaded bitmaps • "font" relates to all downloaded fonts • "xmlui" relates to XML driven user interface available on some platforms[a]
res.quotas.x.value	positive integer	When resources that fall in the defined category are downloaded to the phone, a quota equal to this value * 1024 bytes of compound data size is applied for that category. If downloading a resource would make the quota exceeded for that category, the resource will not be downloaded and a predefined default will be used instead.

a. This is available on MGCP application, but not on SIP application.

4.6.1.25 MicroBrowser <microbrowser/>

These settings control the home page, proxy and size limits to be used by the MicroBrowser when it is selected to provide services.

Attribute	Permitted Values	Default	Interpretation
mb.proxy	Null or domain name or IP address in the format <address>:<port>	Null. Default port = 8080.	Address of the desired HTTP proxy to be used by the MicroBrowser. If blank, normal unproxied HTTP is used by the MicroBrowser.

4.6.1.25.1 Main Browser <main/>

This setting controls the home page used by the MicroBrowser when that function is selected.

Attribute	Permitted Values	Default	Interpretation
mb.main.home	Any fully formed valid HTTP URL. Length up to 255 characters.	Null	URL used for MicroBrowser home-page. If blank, the browser will notify the user that a blank home-page was used. Example: http://www.example.com/xhtml/frontpage.cgi?page=home.

4.6.1.25.2 Browser Limits <limits/>

These settings limit the size of object which the MicroBrowser will display by limiting the amount of memory available for the MicroBrowser.

Attribute	Permitted Values	Default	Interpretation
mb.limits.nodes	Null or positive integer	256	Limits the number of tags which the XML parser will handle. This limits the amount of memory used by complicated pages. A maximum total of 500 (256 each) is recommended. Increasing this value may have a detrimental effect on performance of the phone.

Attribute	Permitted Values	Default	Interpretation
mb.limits.cache	Null or positive integer	200	Limits the total size of objects downloaded for each page (both XHTML and images). Once this limit is reached, no more images are downloaded until the next page is requested. Units = kBytes. Increasing this value may have a detrimental effect on performance of the phone.

4.6.2 Per-phone Configuration - phone1.cfg

This section covers the parameters in the per-phone example configuration file phone1.cfg. This file would normally be used as a template for the per-phone configuration files. For more information, see 2.2.2.1.2 Deploying a Boot Server for the Phones on page 15.

4.6.2.1 Registration <reg/>

SoundPoint® IP 300 and 301 support a maximum of two unique registrations, SoundPoint® IP 500 and 501 support three and SoundPoint® IP 600 and 601 support six. With the attachment of one or more Expansion Modules, the SoundPoint® IP 601 supports an additional six unique registrations, giving a total of twelve. A maximum of three Expansion Modules can be attached. Each registration can optionally be associated with a private array of servers for completely segregated signaling. SoundStation® IP 4000 supports a single registration.

In the following table, x is the registration number. IP 300 and 301: *x*=1-2; IP 500 and 501: *x*=1-3; IP 600: *x*=1-6; IP 601: *x*=1-12; IP 4000: *x*=1.

Attribute	Permitted Values	Default	Interpretation
reg.x.displayName	UTF-8 encoded string	Null	Display name used for local user interface as well as SIP signaling.

Attribute	Permitted Values	Default	Interpretation
reg.x.address	string in the format userPart or userPart@domain	Null	The actual address corresponding to this registration (userPart or userPart@polycom.com). For user part only registration (reg.x.address="1002"), the registration will be userPart@proxyHostNameOrIPAddress where userPart is reg.x.address and proxyHostNameOrIPAddress is either reg.x.server.1.address if non-Null or voIpProt.server.1.address specified in sip.cfg.
reg.x.label	UTF-8 encoded string	Null	Text label to appear on the display adjacent to the associated line key. If omitted, the label will be derived from the user part of reg.x.address.
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 in the same format as reg.x.address	Null	This field must match the reg.x.address value of the other registration which makes up the bridged line.
reg.x.auth.userId	string	Null	User ID to be used for authentication challenges for this registration. If non-Null, will override the "Reg User x" parameter entered into the Authentication submenu off of the Settings menu on the phone.
reg.x.auth.password	string	Null	Password to be used for authentication challenges for this registration. If non-Null, will override the "Reg Password x" parameter entered into the Authentication submenu off of the Settings menu on the phone.

Attribute	Permitted Values	Default	Interpretation
reg.x.server.y.address	dotted-decimal IP address or host name	Null	Optional IP address or host name, port, transport, registration period, fail-over parameters and lineseize subscription period of a SIP server that accepts registrations. Multiple servers can be listed starting with y=1, 2, ... for fault tolerance. If specified, these servers will override the servers specified in sip.cfg in 4.6.1.1.2 Server <server/> on page 71. **Note**: If the reg.x.server.y.address parameter is non-Null, <u>all</u> of the reg.x.server.y.xxx parameters will override the parameters specified in sip.cfg in 4.6.1.1.2 Server <server/> on page 71.
reg.x.server.y.port	0, Null, 1 to 65535	Null	
reg.x.server.y.transport	DNSnaptr or TCPpreferred or UDPonly	DNSnaptr	
reg.x.server.y.expires	positive integer	Null	
reg.x.server.y.register	0, 1	Null	
reg.x.server.y.retryTimeOut	Null or non-negative integer	Null	
reg.x.server.y.retryMaxCount	Null or non-negative integer	Null	
reg.x.server.y.expires.lineSeize	positive integer	Null	
reg.x.acd-login-logout	0, 1	0	If both parameters are set to 1 for a registration, the ACD feature will be enabled for that registration.
reg.x.acd-agent-available	0, 1	0	
reg.x.ringType	1 to 22	2	The ringer to be used for calls received by this registration. Default is the first non-silent ringer.
reg.x.lineKeys	1 to *max*	1	*max* = the number of line keys on the phone. *max* = 1 on SoundStation® IP 4000, *max* = 2 on IP 300 and 301, *max* = 3 on IP 500 and 501, *max* = 6 on IP 600, *max* = 24 on IP 601 (without any Expansion Modules attached, only 6 line keys are available) The number of line keys on the phone to be associated with registration ‘x’.

Attribute	Permitted Values	Default	Interpretation
reg.x.callsPerLineKey	1 to 24 OR 1 to 8	24 OR 8	For the SoundPoint® IP 600 and 601 the permitted range is 1 to 24 and the default is 24. For all other phones the permitted range is 1 to 8 and the default is 8. This is the number of calls or conferences which may be active or on hold per line key associated with this registration. Note that this overrides call.callsPerLineKey for this registration. See 4.6.1.12 Call Handling Configuration <call/> on page 108.

4.6.2.2 Calls <call/>

These sections describe call-oriented per-phone configuration items.

4.6.2.2.1 Do Not Disturb <donotdisturb/>

Attribute	Permitted Values	Default	Interpretation
call.donotdisturb.perReg	0, 1	0	If set to 1, the DND feature will allow selection of DND on a per-registration basis.

4.6.2.2.2 Automatic Off-hook Call Placement <autoOffHook/>

An optional per-registration feature is supported which allows automatic call placement when the phone goes off-hook.

In the following table, x is the registration number. IP 300 and 301: *x*=1-2; IP 500 and 501: *x*=1-3; IP 600: *x*=1-6; IP 601: *x*=1-12; IP 4000: *x*=1

Attribute	Permitted Values	Default	Interpretation
call.autoOffHook.x.enabled	0, 1	0	If set to 1, a call will be automatically placed to the contact specified upon going off hook on this registration.
call.autoOffHook.x.contact	ASCII encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL (6416 or 6416@polycom.com)	Null	

4.6.2.2.3 Missed Call Configuration <serverMissedCall/>

The phone supports a per-registration configuration of which events will cause the locally displayed "missed calls" counter to be incremented.

In the following table, x is the registration number. IP 300 and 301: *x*=1-2; IP 500 and 501: *x*=1-3; IP 600: *x*=1-6; IP 601: *x*=1-12; IP 4000: *x*=1

Attribute	Permitted Values	Default	Interpretation
call.serverMissedCall.x.enabled	0, 1	0	If set to 0, all missed-call events will increment the counter. If set to 1, only missed-call events sent by the server will increment the counter.

4.6.2.3 Diversion <divert/>

The phone has a flexible call forward/diversion feature for each registration. In all cases, a call will only be diverted if a non-Null contact has been configured.

In the following tables, x is the registration number. IP 300 and 301: *x*=1-2; IP 500 and 501: *x*=1-3; IP 600: *x*=1-6; IP 601: *x*=1-12; IP 4000: *x*=1

Attribute	Permitted Values	Default	Interpretation
divert.x.contact	ASCII encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL (6416 or 6416@polycom.com	Null	The forward-to contact used for all automatic call diversion features unless overridden by a specific contact of a per-call diversion feature (see below).
divert.x.autoOnSpecificCaller	0, 1	1	If set to 1, calls may be diverted using the Auto Divert feature of the directory. This is a global flag.
divert.x.sharedDisabled	0, 1	1	If set to 1, all diversion features on that line will be disabled if the line is configured as shared.

4.6.2.3.1 Forward All <fwd/>

Attribute	Permitted Values	Default	Interpretation
divert.fwd.x.enabled	0, 1	1	If set to 1, the user will be able to enable universal call forwarding via the soft key menu.

4.6.2.3.2 Busy <busy/>

Calls can be automatically diverted when the phone is busy.

Attribute	Permitted Values	Default	Interpretation
divert.busy.x.enabled	0, 1	1	If set to 1, calls will be forwarded on busy to the contact specified below.

Attribute	Permitted Values	Default	Interpretation
divert.busy.x.contact	ASCII encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL (6416 or 6416@polycom.com	Null	Forward-to contact for calls forwarded due to busy status, if Null, divert.x.contact will be used.

4.6.2.3.3 No Answer <noanswer/>

The phone can automatically divert calls after a period of ringing.

Attribute	Permitted Values	Default	Interpretation
divert.noanswer.x.enabled	0, 1	1	If set to 1, calls will be forwarded on no answer to the contact specified.
divert.noanswer.x.timeout	positive integer	60	Time in seconds to allow altering before initiating the diversion.
divert.noanswer.x.contact	ASCII encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL (6416 or 6416@polycom.com)	Null	Forward-to contact used for calls forwarded due to no answer, if Null, divert.x.contact will be used.

4.6.2.3.4 Do Not Disturb <dnd/>

The phone can automatically divert calls when DND is enabled.

Attribute	Permitted Values	Default	Interpretation
divert.dnd.x.enabled	0, 1	0	If set to 1, calls will be forwarded on DND to the contact specified below.
divert.dnd.x.contact	ASCII encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL (6416 or 6416@polycom.com)	Null	Forward-to contact used for calls forwarded due to DND status, if Null divert.x.contact will be used.

4.6.2.4 Dial Plan <dialplan/>

Per-registration dial plan configuration is supported. In the following tables, *x* is the registration number. IP 300 and 301: *x*=1-2; IP 500 and 501: *x*=1-3; IP 600: *x*=1-6; IP 601: *x*=1-12; IP 4000: *x*=1

Attribute	Permitted Values	Default	Interpretation
dialplan.x.impossibleMatchHandling	0, 1 or 2	0	When present, and if dialplan.x.digitmap is not Null, this attribute overrides the global dial plan defined in the sip.cfg configuration file. For interpretation, see 4.6.1.2 Dial Plan <dialplan/> on page 77.
dialplan.x.removeEndOfDial	0, 1	1	When present, and if dialplan.x.digitmap is not Null, this attribute overrides the global dial plan defined in the sip.cfg configuration file. For interpretation, see 4.6.1.2 Dial Plan <dialplan/> on page 77.

4.6.2.4.1 Digit Map <digitmap/>

Attribute	Permitted Values	Default	Interpretation
dialplan.x.digitmap	string compatible with the digit map feature of MGCP described in 2.1.5 of RFC 3435; string is limited to 512 bytes and 20 segments; a comma is also allowed; when reached in the digit map, a comma will turn dial tone back on.	Null	When present, this attribute overrides the global dial plan defined in the sip.cfg configuration file. For more information, see 4.6.1.2.1 Digit Map <digitmap/> on page 77.

Attribute	Permitted Values	Default	Interpretation
dialplan.x.digitmap.timeOut	positive integer	Null	When present, and if dialplan.x.digitmap is not Null, this attribute overrides the global dial plan defined in the sip.cfg configuration file. For more information, see 4.6.1.2.1 Digit Map <digitmap/> on page 77.

4.6.2.4.2 Routing <routing/>

This configuration section allows specific routing paths for outgoing SIP calls to be configured independent of other 'default' configuration.

4.6.2.4.2.1 Server <server/>

Attribute	Permitted Values	Default	Interpretation
dialplan.x.routing.server.y.address	dotted-decimal IP address or host name	Null	IP address or host name and port of a SIP server that will be used for routing calls. Multiple servers can be listed starting with y=1, 2, ... for fault tolerance.
dialplan.x.routing.server.y.port	1 to 65535	5060	

4.6.2.4.2.2 Emergency <emergency/>

In the following attributes, *y* is the index of the emergency entry description and *z* is the index of the server associated with the emergency entry *y*. For each emergency entry (index *y*), one or more server entry (indexes (*y*,*z*)) can be configured. *y* and *z* must both follow single step increasing numbering starting at 1.

Attribute	Permitted Values	Default	Interpretation
dialplan.x.routing.emergency.y.value	Comma separated list of entries or single entry representing a or a combination of SIP URL.	Null Example: "15,17,18", "911", "sos".	This represents the URLs that should be watched for emergency routing. When one of these defined URL is detected as being dialed by the user, the call will be automatically directed to the defined emergency server.
dialplan.x.routing.emergency.y.server.z	positive integer	Null	Index representing the server defined in 4.6.2.4.2.2 Emergency <emergency/> on page 136 that will be used for emergency routing.

4.6.2.5 Messaging <msg/>

Message-waiting indication is supported on a per-registration basis.

Attribute	Permitted Values	Default	Interpretation
msg.bypassInstantMessage	0, 1	0	If set to 1, the display offering a choice of "Message Center" and "Instant Messages" will be bypassed when pressing the Messages key. The phone will act as if "Message Center" was chosen. See 3.6.1 Voicemail Integration on page 55. Instant Messages will still be accessible from the Main Menu.

4.6.2.5.1 Message Waiting Indicator <mwi/>

In the following table, *x* is the registration number. IP 300 and 301: *x*=1-2; IP 500 and 501: *x*=1-3; IP 600: *x*=1-6; IP 601: *x*=1-12; IP 4000: *x*=1.

Attribute	Permitted Values	Default	Interpretation
msg.mwi.x.subscribe	ASCII encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL (6416 or 6416@polycom.com)	Null	If non-Null, the phone will send a SUBSCRIBE request to this contact after boot-up.
msg.mwi.x.callBackMode	contact or registration or disabled	"registration" for x = 1, "disabled" for others	If set to "contact", a call will be placed to the contact specified in the callback attribute when the user invokes message retrieval. If set to "registration", a call will be placed using this registration to the contact registered (the phone will call itself). If set to "disabled", message retrieval is disabled.
msg.mwi.x.callBack	ASCII encoded string containing digits (the user part of a SIP URL) or a string that constitutes a valid SIP URL (6416 or 6416@polycom.com)	Null	Contact to call when retrieving messages for this registration.

4.6.2.6 Network Address Translation <nat/>

These parameters define port and IP address changes used in NAT traversal. The port changes will change the port used by the phone, while the IP entry simply changes the IP advertised in the SIP signaling. This allows the use of simple NAT devices that can redirect traffic, but do 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 port 1234.

Attribute	Permitted Values	Default	Interpretation
nat.ip	dotted-decimal IP address	Null	IP address to advertise within SIP signaling - should match the external IP address used by the NAT device.

Attribute	Permitted Values	Default	Interpretation
nat.signalPort	1024 to 65536	Null	If non-Null, this port will be used by the phone for SIP signaling, overriding the value set for voIp-Prot.local.signalPort in sip.cfg.
nat.mediaPortStart	1024 to 65536	Null	If non-Null, this attribute will be used to set the initially allocated RTP port, overriding the value set for tcpIpApp.port.rtp.mediaPortRangeStart in sip.cfg. See 4.6.1.10.3.1 RTP <RTP/> on page 106.

5 Session Initiation Protocol (SIP)

5.1 Basic Protocols

All the basic calling functionality described in the SIP specification is supported. See 5.1.1 RFC and Internet Draft Support on page 141 for supported RFC's and drafts. Transfer is included in the basic SIP support.

5.1.1 RFC and Internet Draft Support

ID	Title
RFC 2387	The MIME Multipart / Related Content-type
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 3515	The Session Initiation Protocol (SIP) Refer Method
draft-ietf-sip-cc-transfer-05.txt	SIP Call Control - Transfer
draft-ietf-sip-replaces-03.txt	The Session Initiation Protocol (SIP) "Replaces" Header

5.1.2 Request Support

Method	Supported	Notes
REGISTER	Yes	
INVITE	Yes	
ACK	Yes	
CANCEL	Yes	
BYE	Yes	

Method	Supported	Notes
OPTIONS	Yes	
SUBSCRIBE	Yes	
NOTIFY	Yes	
REFER	Yes	
PRACK	Yes	

5.1.3 Header Support

In the following table, a “Yes” in the Supported column means the header is sent and properly parsed.

Header	Supported	Notes
Accept	Yes	
Accept-Encoding	No	
Accept-Language	No	
Alert-Info	Yes	
Allow	Yes	
Allow-Events	Yes	
Authentication-Info	No	
Authorization	Yes	
Call-ID	Yes	
Call-Info	Yes	
Contact	Yes	
Content-Disposition	No	
Content-Encoding	No	
Content-Language	No	
Content-Length	Yes	
Content-Type	Yes	
CSeq	Yes	
Date	No	
Diversion	Yes	

Header	Supported	Notes
Error-Info	No	
Event	Yes	
Expires	Yes	
From	Yes	
In-Reply-To	No	
Max-Forwards	Yes	
Min-Expires	No	
Min-SE	Yes	
MIME-Version	No	
Organization	No	
P-Asserted-Identity	Yes	
P-Preferred-Identity	Yes	
Priority	No	
Proxy-Authenticate	Yes	
Proxy-Authorization	Yes	
Proxy-Require	No	
RAck	Yes	
Record-Route	Yes	
Refer-To	Yes	
Referred-By	Yes	
Remote-Party-ID	Yes	
Replaces	Yes	
Reply-To	No	
Require	Yes	
Retry-After	No	
Route	Yes	
RSeq	Yes	
Server	No	
Session-Expires	Yes	
Subject	No	
Subscription-State	Yes	

Header	Supported	Notes
Supported	Yes	
Timestamp	No	
To	Yes	
Unsupported	No	
User-Agent	Yes	
Via	Yes	
Warning	No	
WWW-Authenticate	Yes	

5.1.4 Response Support

In the following table, a “Yes” in the Supported column means the header is parsed. The phone may not actually generate the response.

5.1.4.1 1xx Responses - Provisional

Response	Supported	Notes
100 Trying	Yes	
180 Ringing	Yes	
181 Call Is Being Forwarded	No	
182 Queued	No	
183 Session Progress	Yes	

5.1.4.2 2xx Responses - Success

Response	Supported	Notes
200 OK	Yes	
202 Accepted	Yes	In REFER transfer.

5.1.4.3 3xx Responses - Redirection

Response	Supported	Notes
300 Multiple Choices	Yes	
301 Moved Permanently	Yes	
302 Moved Temporarily	Yes	
305 Use Proxy	No	
380 Alternative Service	No	

5.1.4.4 4xx Responses - Request Failure

All 4xx responses for which the phone does not provide specific support will be treated the same as 400 Bad Request.

Response	Supported	Notes
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	No	

Response	Supported	Notes
480 Temporarily Unavailable	Yes	
481 Call/Transaction Does Not Exist	Yes	
482 Loop Detected	Yes	
483 Too Many Hops	No	
484 Address Incomplete	Yes	
485 Ambiguous	No	
486 Busy Here	Yes	
487 Request Terminated	Yes	
488 Not Acceptable Here	Yes	
491 Request Pending	No	
493 Undecipherable	No	

5.1.4.5 5xx Responses - Server Failure

Response	Supported	Notes
500 Server Internal Error	Yes	
501 Not Implemented	Yes	
502 Bad Gateway	No	
503 Service Unavailable	No	
504 Server Time-out	No	
505 Version Not Supported	No	
513 Message Too Large	No	

5.1.4.6 6xx Responses - Global Failure

Response	Supported	Notes
600 Busy Everywhere	No	
603 Decline	Yes	
604 Does Not Exist Anywhere	No	

Response	Supported	Notes
606 Not Acceptable	No	

5.1.5 Hold Implementation

The phone supports both currently accepted means of signaling hold. The first method, no longer recommended due in part to the RTCP problems associated with it, is to set the "c" destination addresses for the media streams in the SDP to zero, for example, c=0.0.0.0. The second, and preferred, method is to signal the media directions with the "a" SDP media attributes sendonly, recvonly, inactive or sendrecv. The hold signaling method used by the phone is configurable (for more information, see 4.6.1.1.3 SIP <SIP/> on page 73) but both methods are supported when signaled by the remote endpoint.

5.1.6 Reliability of Provisional Responses

The phone fully supports RFC 3262 - *Reliability of Provisional Responses*.

5.1.7 Transfer

The phone supports transfer using the REFER method specified in draft-ietf-sip-cc-transfer-05 and RFC 3515.

5.1.8 Third Party Call Control

The phone supports the delayed media negotiations (INVITE without SDP) associated with third party call control applications.

5.2 Protocol Extensions

The phone supports the following SIP protocol extensions.

5.2.1 RFC and Internet Draft Support

ID	Title
RFC 1321	The MD5 Message-Digest Algorithm
RFC 3311	The Session Initiation Protocol (SIP) UPDATE Method
RFC 3325	SIP Asserted Identity
RFC 3725	Best Current Practices for Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP)
draft-levy-sip-diversion-04.txt	Diversion Indication in SIP
draft-ietf-sip-session-timer-12.txt	Session Timers in the Session Initiation Protocol (SIP)
draft-ietf-sipping-mwi-02.txt	A Message Summary and Message Waiting Indication Event Package for the Session Initiation Protocol (SIP)
draft-ietf-sipping-dialog-package-03.txt	INVITE Initiated Dialog Event Package for the Session Initiation Protocol (SIP)
draft-ietf-sip-privacy-04.txt	SIP Extensions for Network-Asserted Caller Identity and Privacy within Trusted Networks
draft-ietf-sip-referredby-05.txt	SIP Referred by Mechanism
draft-levy-sip-diversion-06.txt	Diversion Indication in SIP
draft-ietf-sipping-cc-conferencing-03.txt	SIP Call Control - Conferencing for User Agents

5.2.2 Request Support

Method	Supported	Notes
INFO	Yes	RFC 2976, the phone does not generate INFO requests, but will issue a final response upon receipt. No INFO message bodies are parsed.
MESSAGE	Yes	Final response is sent upon receipt. Message bodies of type text/plain are sent and received.
UPDATE	Yes	

5.2.3 SIP for Instant Messaging and Presence Leveraging Extensions (SIMPLE)

The phone is compatible with the Presence and Instant Messaging features of Windows® Messenger® and MSN® Messenger 4.7 and Windows® Messenger® 5.0. In a future release, support for the Presence and Instant Message recommendations in the SIP *SIMPLE* proposals will be provided:

- draft-ietf-simple-cpim-mapping-01
- draft-ietf-simple-presence-07
- draft-ietf-simple-presencelist-package-00
- draft-ietf-simple-winfo-format-02
- draft-ietf-simple-winfo-package-02

or their successors.

5.2.4 Shared Call Appearance Signaling

A shared line is an address of record managed by a server. The server allows multiple endpoints to register locations against the address of record.

The phone supports shared call appearances (SCA) using the SUBSCRIBE-NOTIFY method in the "SIP Specific Event Notification" framework (RFC 3265). The events used are:

- "call-info" for call appearance state notification
- "line-seize for the phone to ask to seize the line

5.2.5 Bridged Line Appearance Signaling

A bridged line is an address of record managed by a server. The server allows multiple endpoints to register locations against the address of record.

The phone supports bridged line appearances (BLA) using the SUBSCRIBE-NOTIFY method in the "SIP Specific Event Notification" framework (RFC 3265). The events used are:

- "dialog" for bridged line appearance subscribe and notify

6 Appendix 1

6.1 Trusted Certificate Authority List

The following certificate authorities are trusted by the phone by default.

ABAecom (sub., Am. Bankers Assn.) Root CA

ANX Network CA by DST

American Express CA

American Express Global CA

BelSign Object Publishing CA

BelSign Secure Server CA

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

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

National Retail Federation by DST

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

UPS Document Exchange by DST

ValiCert Class 1 VA

ValiCert Class 2 VA

ValiCert Class 3 VA

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 4 Public Primary Certification Authority - G2

Verisign Class 4 Public Primary Certification Authority - G3

Verisign/RSA Commercial CA

Verisign/RSA Secure Server CA

7 Appendix 2

7.1 Third Party Software Attribution

The following third party software products are part of the SIP application.

Ares:

Copyright 1998 by the Massachusetts Institute of Technology.
Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.
M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

OpenSSL

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.
OpenSSL License
Copyright (c) 1998-2003 The OpenSSL Project. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:
"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse

or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"
THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).
Original SSLeay License:
Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
All rights reserved.
This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscape's SSL.
This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).
Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display

the following acknowledgement: "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)"
The word 'cryptographic' can be left out if the routines from the library being used are not cryptographic related.
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
The licence and distribution terms for any publicly available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

zlib:

(C) 1995-2002 Jean-loup Gailly and Mark Adler
This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.
Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:
1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.
Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

Expat:

Copyright (c) 1998, 1999, 2000 Thai Open Source Software Center Ltd and Clark Cooper

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

curl:

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2004, Daniel Stenberg, <daniel@haxx.se>.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.