

UC Software Provisioning Best Practices

By Marek Dutkiewicz

Director, Polycom VoIP Product Management

The information presented in this white paper applies to administrators who manage and provision Polycom phones using Polycom Unified Communications (UC) Software release versions 3.3.0 with SoundPoint IP, SoundStation IP, and VVX 1500 phone models deployed in a large-scale enterprise environment. For phones using SIP 3.2.5 or earlier, see [Configuration File Management on Polycom® SoundPoint® IP, SoundStation® IP, and VVX® Phones](#).

This white paper applies to administrators provisioning SoundPoint IP, SoundStation IP, and VVX 1500 phone models using UC software release versions 3.3.0 or later.

Introduction

The best practices and recommendations we will be describing in this white paper will assist you to deploy and manage Polycom VoIP devices in a variety of deployment scenarios. In particular, this white paper will illustrate how to deploy phones in complex environments such as those seen in large-scale, multi-tenanted, hosted VoIP services.



Download the [Administrator's Guide](#) for an explanation of the configuration parameters that relate to provisioning operations. Many of the provisioning functions discussed in this white paper refer to the *Administrator Guide*.

The term 'provisioning' describes the process of configuring a device (such as a phone) to interact with a network infrastructure to deliver the functionality and user experience that has been determined by the system administrator. In the case of Polycom VoIP devices, this involves:

- Specifying the settings required for the device to properly interact with the VoIP call server and the associated network infrastructure
- Enabling the relevant features required on the device
- Configuring the user interface (as decided by the system administrator)
- Enabling certain device features that may require a feature license 'key'
- Specifying security settings to prevent malicious or unauthorized use

You can control the many features and functions of Polycom devices by provisioning the configuration parameters using one of the following methods:

- Using centralized configuration files
- From the phone's web interface
- From the phone's native keypad/LCD interface

Polycom strongly recommends using centralized configuration files for installations of 20 or more devices, and/or for installations that harness advanced features on the device. For installations involving less than 20 devices you have the option of using the Polycom Web Configuration Utility or the phone's local interface.

Keep in mind that configuring settings locally using the phone keypad or Web Configuration Utility will override any configuration settings that you apply from a central server, making it difficult for you to manage configurations across all devices in a large-scale device deployment from a central server.

If you are using a centralized provisioning system to manage your devices, Polycom recommends that you take action to prevent end users from accessing their phone configuration locally using both of the following methods:

- Change the administrator password to prevent end users from performing administrator changes using the keypad interface, and;
- Disable access to the phone's Web Configuration Utility by setting `httpd.cfg.enabled="0"` in the configuration files.

Understanding the Phone Software Components

Before we get too far into provisioning, let's examine each of the components that comprise the phone software and explore what happens during each of the provisioning stages described in the preceding section.

The phone is comprised of two key software components launched in succession, as follows:

- **BootROM Software**

First, the phone loads the BootROM software into memory. This is a small software component upgraded in a fully fault-tolerant fashion. During a BootROM update, the old version remains intact. This enables the phone to tolerate an interruption (such as a power outage or link breakage) without requiring an RMA (Return Materials Authorization) to address such a failure. BootROM software is designed to establish network connectivity, discover the provisioning server, and upload either a new BootROM version or the main UC software application as needed.

- **Polycom UC Software**

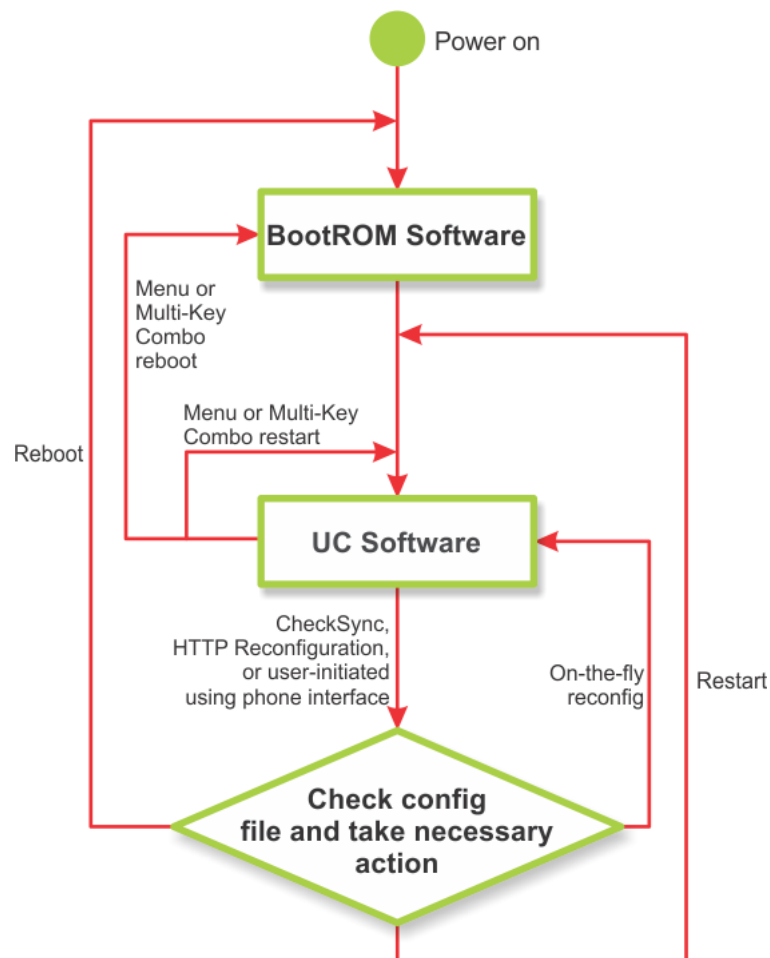
Once the BootROM phase is complete, the phone loads the UC software application to set up the product features and functionality. Due to memory constraints, only one version of the UC software is stored on the device. If the UC software upgrade fails, the BootROM software has the ability to load a new UC software application version.

With the exception of the VVX 1500, there are very few software interdependencies between the BootROM and UC software applications. This means you can independently manage your application upgrades for these files. For information on exceptions to this, refer to the software release notes provided with the upgrade.



Polycom plans to combine the BootROM and UC Software applications into a single bundle for all products starting with the release of UCS 4.0.0 scheduled for mid-2011.

The illustration shown next depicts each of the provisioning lifecycle phases discussed in this white paper:



What Does Centralized Provisioning Enable You to Do?

Centralized provisioning enables you to apply configuration settings to multiple devices and device types within your deployment. The SoundPoint IP, SoundStation IP, and VVX 1500 phones support several transport protocols for provisioning, including HTTPS, HTTP, FTP, and TFTP.

You can use one or more servers in combination with any of these protocols to manage the following:

- **Configuration Files (Read Only)** Manage which configuration files control phone behavior.
- **Overrides Files – User Preferences** Manage override settings for changes made using the phone’s user interface. This includes both user preferences, such as ringtones, language settings, and custom backgrounds, as well as any administrator changes.
- **Web Overrides** Manage changes made to the phone configuration using the browser-based Web Configuration Utility.
- **Contact Directory Back-Up** Manage the location of your contact directory backup files. For more information, see *Configuring Your Local Contact Directory* discussed later in this white paper.
- **Log Files** Manage when and where log files are stored. Log files store information about the phone’s operation that may be useful for troubleshooting/support purposes.
- **License Files** Manage the location where license files required to enable certain features on the device are downloaded from.

When Does Phone Provisioning Occur?

Phones are generally provisioned during the phone startup phase, which can occur because of any of the following events:

Powering Up the Phone

When you provide power to an unpowered phone, the phone will follow a series of power-up events. As the phone starts up, it reads the configuration files on the central provisioning server for its configuration settings.

Rebooting the Phone

When you perform a reboot, the phone behaves as though it were powering up. You can reboot a phone using the keypad interface by choosing Menu > Settings > Advanced, entering the administrator password, and choosing Reboot Phone. Pressing the appropriate Multi-Key Combo (MKC) on the keypad interface also causes the phone to reboot. For more information on using multi-key combinations, see [Restarting and Resetting SoundPoint IP, SoundStation IP and Polycom VVX 1500 Phones, Quick Tip 18298](#).

Restarting the Phone

When you perform a restart, the phone reads the configuration files and loads the UC software application, but does not load the BootROM software. You can restart a phone using the keypad interface by choosing Menu > Settings > Basic > Restart Phone. Since the process excludes the BootROM, a restart is faster than a reboot while still provisioning the phone.

Reconfiguring the Phone Locally

When you make changes to certain phone settings, the phone will automatically restart or reboot to effect the changes. You can also manually trigger a configuration update using either the keypad interface (by choosing Menu > Settings > Basic > Update Configuration) or the Web Configuration Utility (by choosing Utilities > Software Upgrade). In both instances, the phone will re-read the configuration files and apply the settings. The advantage of this method over restart or reboot is that the phone will apply changes to multiple settings on-the-fly. For more information on which parameter changes trigger a reboot or restart, refer to your Administrator Guide.

When making configuration changes through the Web Configuration Utility, the phone will initiate a reconfiguration, restart, or reboot depending on the settings that you have changed.

Reconfiguring the Phone Remotely

You can initiate a reconfiguration by sending a Check-Sync SIP NOTIFY message to the phone. This will cause the phone to compare its cached software version and configuration file to the versions on the provisioning server. The phone will apply any required changes on-the-fly or through a reboot, depending on the parameter being changed. See your Administrator's Guide for information on which parameter changes trigger an automatic reboot or restart.

In the example shown next, a Check-Sync SIP NOTIFY message is used to initiate provisioning. This is detailed in the attached 'Wireshark packet capture' (simpleCheckSync.pcap).



```
From: <sip:akihabara>;tag=1530231855-106746376154
To: <sip:192.168.1.104:5060>
Call-ID: b98cd3c4-f37c265b-c864532
CSeq: 1 NOTIFY
Contact: <sip:akihabara>
Event: check-sync
Max-Forwards: 10
Content-Length: 0
```

In the UCS 3.3.1 release, Polycom added an HTTP-Reconfig option to initiate a re-configuration option using the Push API.

An example of how this is used is as follows:

```
<html xmlns="http://www.w3.org/1999/xhtml">
  <p>Reconfigure example </p><br/>
  <a href="Action:UpdateConfig">Reconfig</a><br/>
</html>
```

Scheduling Automatic Phone Reconfigurations

You can configure the phones to check their configuration automatically according to a set schedule. For example, you can set your phones to check for configuration changes each day at a specific time. For information on setting these options, download the Administrator's Guide, locate to the *Provisioning <polling/>* table, and refer to the *prov.polling* attribute.



Since the polling feature is disabled by default, you will need to enable it during your initial configuration before you can schedule the phones to automatically check their configuration.

Discovering the Provisioning Server

After you reboot or restart a phone, it searches for its provisioning server using DHCP or the URL or IP address configured in Menu->Settings->Advanced->Network Configuration->Server Menu.

The steps that your phone follows to discover the server are controlled by the Dynamic Host Configuration Protocol (DHCP) settings associated with the provisioning/boot server.

This section describes the factory default process that your phone follows for discovering the provisioning server. Keep in mind that you can override these settings by changing the boot server setting through the Administrator settings section.

During the server-discovery process, your phone follows these steps:

- **Stage 1: Phone Receives an Option Setting in a DHCP Response**

The phone will request an IP address using the DHCP protocol. The resulting response may contain *option 66* or 160 (by factory default) that contains the provisioning server address. If so, the phone will use it as the provisioning server address.



You can configure some DHCP servers to check Option 60 for information on the device and respond with an option specific to the Polycom phones. For more information, see [Using DHCP Vendor Identifying Options With Polycom SoundPoint IP, SoundStation IP and VVX Phones, Technical Bulletin TB54041](#).

- **Stage 2: Phone Sends a DHCP INFORM Request**

If the initial response from the DHCP server does not contain an address in Option 66 or the Custom Option, the phone will send out a DHCP INFORM request asking for these options. A server other than the main DHCP server may respond with the provisioning server address.



Currently, a phone will only send a DHCP INFORM request if the boot/provisioning server address is set to 0.0.0.0 or <null>. In a future software release, Polycom plans to extend this request to include the ztp.polycom.com address. In cases where a valid boot server address is configured, the phone will not send a DHCP INFORM request. This speeds the boot-up time when non-static boot server settings are used in the DHCP menu.

- **Stage 3: Phone Uses Configured Static Boot Server**

If neither of the preceding two discovering methods result in a provisioning server address, or you configure the phone to use a Static boot server (Boot Server: Static), then the phone will use the statically configured address. The factory default for this address is currently <null>. It had previously been set to 0.0.0.0 in manufacturing. As of March 2011, Polycom's factory setting will change this default address to <https://ztp.polycom.com>. You can reset the phone to factory defaults using the keypad interface by choosing Settings > Advanced > Administration Settings > Reset to Defaults.

- **Stage 4: Phone Uses Its Internally Cached Configuration**

If none of the previously described operations provide a reachable provisioning server, the phone will boot up using its internally cached configuration and the phone will display the following temporary message: *Could not contact boot server using existing configuration*.



The BootROM and UC software application will each independently perform the provisioning discovery process during their start-up. This means that in the case of a reboot, the phone will perform this operation twice.

The settings of the Boot Server options can greatly affect boot up speed. For more information, see [Optimizing Boot Speed of Polycom SoundPoint IP products, Technical Bulletin 41856](#).

Understanding Phone Configuration Parameters

Phone behavior and functionality is set using a wide range of configuration parameters. These parameters control everything from startup behavior to default user preferences. Let's take a close look at what types of configuration parameters are available.



For the complete list of phone configuration parameters, functions, and values that may be set, see your Administrator's Guide. For general information on formatting and using configuration parameters, see [Simplified Configuration Improvements in Polycom® UC Software 3.3.0, Technical Bulletin 60519](#).

Configuring Device Settings in Flash Memory

Device settings are configuration parameters that set network and server addresses, password information, as well as security certificates. Device settings are stored in a protected area of non-volatile memory (also known as *flash* memory) making the settings immune to erroneous changes or data corruption. The following configuration parameters are common examples of these settings, which can also be set locally using the keypad interface:

- Network settings related to layer 2 access, such as Link Layer Discovery Protocol (LLDP), Virtual Local Area Network (VLAN), Ethernet link negotiation, DHCP, Domain Name Server (DNS), and Server Network Time Protocol (SNTP)
- Provisioning server address and configuration
- User and administrator passwords
- Network configuration settings
- Custom Certification Authority (CA) security certificate

If needed, you can set these device settings using the central configuration files. This is a process known as *device.set* and requires that you carefully pre-plan to guard against conflicting parameter settings between the central and local settings. Some errors, such as statically setting the phone's IP address, may require you to change the local settings on each affected phone.



For information on performing device setting changes, see the <device/> configuration parameter defined in Appendix A: Configuration Files of your Administrator's Guide.

Certain configuration parameters, such as the SNTP server settings, may be set in both *device settings* and *normal* configuration files. In these situations, you can use a configuration parameter to set which settings the phone will use.

Configuring Your Flash File System Settings

Configuration parameters that do not fall into the *device settings* category are stored using a *Flash File System*. These settings are stored on the phone even after it powers down. However, this storage method is not as robust as the device settings storage. For example if you upgrade the phone's BootROM, these parameters will be erased.



On the VVX1500 phone, a software upgrade does not erase the Flash File system. This same behavior is planned for the upcoming UCS 4.0.0 release, which will treat the BootROM and application as a single image.

Configuring Your Local Contact Directory

The phone supports a user-defined contact directory. This directory is maintained locally on the phone in the Flash file system and is read and backed-up to the provisioning server path specified in the CONTACTS field of the <MAC-ADDRESS>.cfg or 000000000000.cfg file.

During each boot-up phase, the phone will read the file from the provisioning server and update the local copy based on the results. If the end user changes, adds, and/or removes a directory contact using the phone keypad or web interface, the phone will attempt to create a new copy of the contact directory file at the address specified for the provisioning server.

The phone reads this directory file during a reboot and/or restart. However, the phone does *not* read the file on an Update Configuration/CheckSync/HTTP reconfig event unless the permission on the directory file is set as readonly `dir.local.readonly="1"`.

There is one special exception worthy of description. If the phone does not have a local contact directory present in its internal memory AND there is no <MAC-ADDRESS>-directory.xml file on the provisioning server, the phone will attempt to read a 000000000000-directory.xml file. You can use this method to pre-populate a contact directory with all phones in a deployment. Note that the phone will save a copy of this file internally when it is read such that the 000000000000-directory.xml will not be used unless the contact directory is subsequently cleared by a Flash File System Format operation.



Be certain to exercise caution when making changes to the local directory file to avoid the following unexpected results:

- The administrator edits the directory file on the server causing the phone to reboot or to reconfigure and update (over-write) the directory.
- The provisioning server permissions do not allow write access by the user, causing any directory changes to be lost when a reboot or restart the phone.
- Be certain that you have set the access permissions on the Contact Directory to read-only to prevent the local phone directory from changing the directory. This is useful if you are using a utility to manipulate the directory file on the server and you need to prevent local changes that may conflict with directory file changes.

Resetting Parameters to Defaults

If you need to reset a phone's configuration settings to defaults for whatever reason, you can use the Reset Local Config option. This will remove all the settings made by the user for items such as ringtones, background selections, and so on. You can use this command to reset a phone to defaults for a new user.

When you use the Reset Local Config option, the phone will attempt to write an empty override file <MAC-ADDRESS>-phone.cfg to the location specified by the OVERRIDES_DIRECTORY configuration setting in the Master Configuration file. If the phone can't override the files, and there is an override file present on the provisioning server, the old settings will be reapplied.

Resetting Software to Factory Defaults

Polycom ships phones from the factory with a consistent, documented set of defaults for the software version and default device settings. Polycom occasionally updates the shipping configuration, which can vary between phone models.

Several months prior to any shipping changes, Polycom will release a technical bulletin describing the factory default settings. This information includes the effective date and MAC address details. For examples of how these changes appear, see the following technical bulletins:

- [*Product Shipping Configuration Change Notice, Technical Bulletin 64309*](#);
- [*SIP Factory Configuration Change on SoundPoint IP Phones, Technical Bulletin 38447*](#), and;
- [*SIP Factory Configuration Change on SoundPoint IP and SoundStation IP Phones, Technical Bulletin 46440*](#)



If you are an administrator resetting or updating a device to factory default settings, be aware that the device settings on the phone following a 'reset to defaults' may differ depending on the phone's manufacture date.

Keep in mind that resetting a phone to factory defaults may involve multiple steps. Depending on exactly which parameters need to be reset, you may need to follow one or more of the following steps:

- 1 Choose Menu > Settings > Advanced > Admin Settings > Reset to Defaults and individually choose each of the following:
 - Reset Local Configuration
 - Reset Web Configuration
 - Reset Device Settings
 - Format File System
2. Delete any override files that are located on the provisioning server.



Keep in mind that choosing the Format File System operation will remove the UC Software, causing the phone to reload the software during the next provisioning cycle.



An enhancement planned for the upcoming UC Software 4.0.0 release will consolidate the full factory reset into a single step.

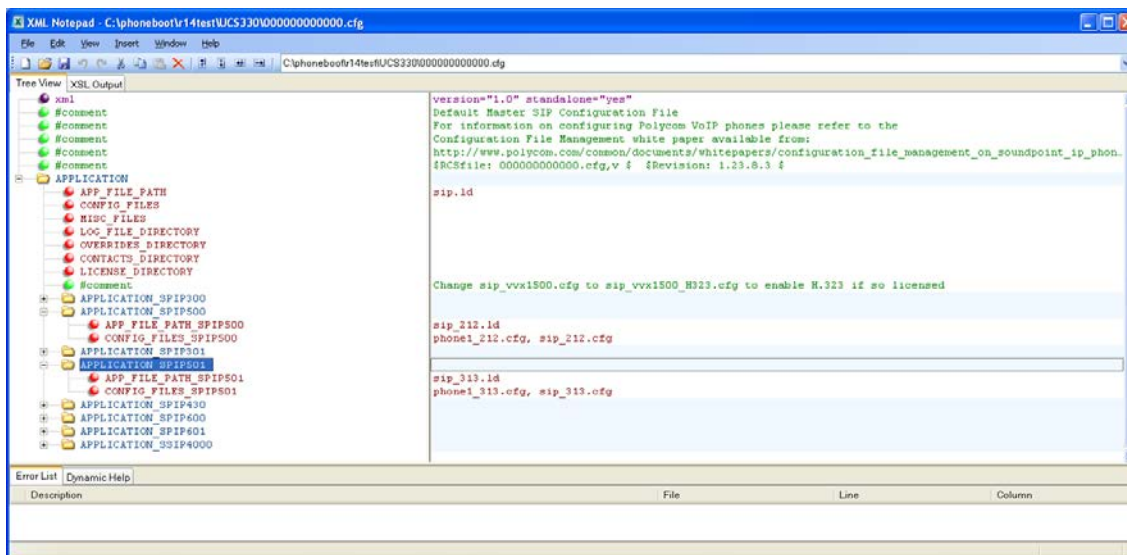
Configuration File Details

This section describes various aspects of the configuration file format, structure, and usage. Configuration file details include XML attributes, security, phone template files, default configuration parameter values, CMA provisioning, and software version.

Master Configuration File

The *Master* configuration file controls several aspects of the phone's configuration and management functions. You can deploy the master configuration file on a per-phone basis using the <MACADDRESS>.cfg file, or as a master version using the 000000000000.cfg file. To ease administration, Polycom recommends using the 000000000000.cfg version. When doing so, you can use wildcard substitutions to control certain parameters on a per-phone by using the PHONE_MAC_ADDRESS keyword, or a per-model basis using the PHONE_MODEL_NUMBER keyword. For details on how to use these settings, refer to your Administrator's Guide or see [Overriding Parameters in Master Configuration Files on SoundPoint IP and SoundStation IP Phones, Technical Bulletin 35361](#). Note that the substitution string needs to be enclosed in [] in the configuration file e.g. [PHONE_MAC_ADDRESS]

An example of a 000000000000.cfg file is included with each release distribution. These attributes are stored using XML structure as shown next:



Each configuration file includes configuration attributes used by the phone. These attributes are stored using XML structure and defined as follows:

- **APP_FILE_PATH** This attribute is a pointer to the software image file. By default, the file name is sip.ld. However, if you rename the file on the server, you must also rename it in the config file. If this field is set to null, the phone will not check for a software file.

- **CONFIG_FILES** This is a comma-separated list of configuration file values that the phone should read during configuration. These files may each be located in different sub-directories or even on different servers using different protocols. Standard URL naming rules are applied to user names and passwords in the address. If the phone encounters configuration parameters in multiple files, the leftmost files take precedence.
- **MISC_FILES** This specifies the location that the phone should look for certain files relating to the phone operation and/or configuration. This parameter is historical in nature and provides no useful function in current software releases.
- **LOG_FILE_DIRECTORY** This value specifies the location where the phone will write the log files generated during phone operation. This location may be the same as the provisioning server or a completely different server, and may use a different transport protocol if needed. If this parameter is set to null, the phone will write the logs to the root of the provisioning server path as long as the phone has write access permission.
- **OVERRIDES_DIRECTORY** This location defines where the phone will read and/or write the override settings. The override settings are values that have been changed locally using the phone keypad or web server interface. These include both user settings, such as ring tone choices, as well as Administrator settings made from the phone UI. This location may be the same as the provisioning server or a completely different server and may use a different transport protocol if needed.

If this parameter is set to null, the phone will read and/or write the overrides to the root of the provisioning server path as long as the phone has write access permission. If the **OVERRIDES_DIRECTORY** is not set to write access, the phone will use the internally cached version of this file.

If the Flash File System is formatted, or a Reset Local Config operation is performed OR a BootROM upgrade is carried out, the local copies of the overrides (<MAC-ADDRESS>-phone.cfg) and web overrides (<MAC-ADDRESS>-web.cfg) files will be lost and the user's preferences will revert back to the default setting if a copy has not been saved on the provisioning server.
- **CONTACT_DIRECTORY** This is the location where the phone will read and/or write the local contact directory file. The contact directory file includes all entries in the local contact directory, including settings such as speed dials, specialized ringtone treatment, and so on. The phone will rewrite this file whenever a contact is added, changed, or deleted. The system will only update the local phone directory with changes made to the server contact directory if the phone reads the server file first. If you wish to use the local contact directory as the server-based directory, Polycom recommends that you set the directory file with read-only permissions.

- If this parameter is set to null, the phone will read and/or write the Contact Directory (<MAC-ADDRESS>-directory.xml) to the root of the provisioning server path as long as the phone has write access permission. If the CONTACT_DIRECTORY does not allow the phone write access, the phone will use the internally cached version of this file.

The local copy of the Contact Directory will be lost if any of the following occur:

- » The Flash File System is formatted, or;
- » A BootROM upgrade is carried out.

If a copy has not been saved on the provisioning server the Contact Directory will read the 000000000000-directory.xml file or revert to an empty file (if there is no <MAC-ADDRESS>-directory.xml) or 000000000000-directory.xml file present.

- **LICENSE_DIRECTORY** This value specifies the location that the phone will look for license files to enable licensed features such as the Productivity Suite. The phone will look for both a phone-specific license in the <MACADDRESS>-license.cfg file and a site license in the 000000000000-license.cfg file. If this parameter is set to null, the phone will look in the root directory of the provisioning server. For more details on the use of license files, see [Best Practices When Upgrading License Keys on SoundPoint IP and SoundStation IP Phones, Technical Bulletin 44210](#).

Configuration File Security

Polycom recommends using the https protocol using mutual authentication for downloading and uploading all provisioning-related files. See the Administrator's Guide for more details.

In situations where https is not used, configuration file encryption may be used to provide file security. This feature requires a shared key to be loaded on the phone and typically requires performing a manual implementation using a secure environment (configuration file encryption is therefore less desirable than HTTPS). Since the Encryption SDK contains software that is subject to export control laws, request this directly from Polycom technical support either through a registered Polycom distributor or reseller, or by submitting an online request at http://www.polycom.com/forms/api_dev_reg.html.

Phone Template Files

Each software release contains a set of template configuration files. You can use these files as a reference to document the exact syntax and default values for each parameter. Polycom recommends that you do not use these files directly. Rather, Polycom provides these as examples of parameter settings for you to cut and paste into the configuration files managed by the provisioning system. You will only need to do this for parameters that differ from the defaults.

Default Configuration Parameter Values

The software releases have built-in default values for each configuration parameter. You will find the values for each configuration parameter defined in the template files included with the release and in the Administrator's Guide. In the event of any differences, treat the template files as the definitive reference.

Many parameters on the phone may be set to non-default values using more than one method. In the event that the same parameters are set using more than one method, the order of precedence is as follows:

- 1. Keypad interface (read from server) - <MAC-ADDRESS>-phone.cfg**

If this file is not found on the server, the phone will use the locally cached version of this file.

- 2. Web interface (read from server) - <MAC-ADDRESS>-web.cfg**

If this file is not found on the server, the phone will use the locally cached version of this file.

- 3. Configuration files (read from server) as listed in the master configuration file**

If *one or more* files could not be read from the provisioning server, the phone will use the cached version for all configuration files listed in the master configuration.

If the server versions of any of the configuration files are not found during a reboot, the INTERNAL copy will be used instead (as detailed above). This can occur for one of the following reasons:

- The provisioning server is unavailable during boot.
- The override files were not written to the provisioning server. For example, this can occur if the server does not have write access permissions, or the phone was pre-staged on a server that is different from the one it is currently accessing.

If the same parameter is set in multiple configuration files, the file read first (left to right) will take precedence. As a general rule, the phone handles precedence using a top to bottom, left to right ranking.

CMA Provisioning

With the release of UCS version 3.3.1, you can provision and manage the VVX 1500 phone using the Polycom CMA solution. Although the details of this solution are beyond the scope of this white paper, see Polycom's eSupport site at <http://support.polycom.com/> for more information.

Managing Different Software Versions on a Network

Managing a variety of Polycom phone models in a typical large-scale deployment requires that you pay close attention to the software versions used by each model type. You must take precautions to ensure that you correctly match the phone configuration files to the corresponding phone models.

The reason for this stems from the limitations on older phones. Due to memory limitations, certain older phone models may not support the features available in the newest software versions. Polycom continues to support these phones through updates to older software versions.

For more information on how to manage different software versions in a single network, see [Supporting Legacy Polycom Phones with Later Releases, Technical Bulletin 35311](#).

Understanding Phone Update Behavior

Use the following table as a guide to the different methods of applying configuration changes, software updates, and contact directories:

Action	Config Parameter Change	Software Update	Contact Directory Update
Reboot (Menu, MKC or power cycle)	Yes	Yes	Yes
Restart (Menu or MKC)	Yes	Yes (see Note 1)	Yes
Update Config (Menu)	Yes	Yes	Only if dir.local.readonly="1"
SIP NOTIFY (Check-Sync)	Yes	Yes	Only if dir.local.readonly="1"
HTTP Reconfig	Yes	Yes	Only if dir.local.readonly="1"
Config. Polling	Yes	Yes	Only if dir.local.readonly="1" -



The restart operation takes a while to discover that it needs to perform a software upgrade and is slower than executing a reboot or reconfiguration operation.

Additional Information

Polycom recommends that you download the Administrator's Guide from Polycom's eSupport site at http://www.polycom.com/support/voice/soundpoint_ip/soundpoint_ip670.html.

The following is a list of technical bulletins referenced in this document and includes relevant quick tips. You can also download each of these documents from the Polycom eSupport Web site at: http://www.polycom.com/support/voice/soundpoint_ip/VoIP_Technical_Bulletins_pub.html.

[Using DHCP Vendor Identifying Options with Polycom SoundPoint IP, SoundStation IP and VVX Phones, Technical Bulletin TB54041](#)

[Optimizing Boot Speed of Polycom SoundPoint IP products, Technical Bulletin 41856](#)

[Simplified Configuration Improvements in Polycom® UC Software 3.3.0, Technical Bulletin 60519](#)

[SIP Factory Configuration Change on SoundPoint IP Phones, Technical Bulletin 38447](#)

[SIP Factory Configuration Change on SoundPoint IP and SoundStation IP Phones, Technical Bulletin 46440](#)

[Overriding Parameters in Master Configuration Files on SoundPoint IP and SoundStation IP Phones, Technical Bulletin 35361](#)

[Best Practices When Upgrading License Keys on SoundPoint IP and SoundStation IP Phones, Technical Bulletin 44210](#)

[Supporting Legacy Polycom Phones with Later Releases, Technical Bulletin 35311](#)

[Redundant Boot Servers for SoundPoint IP Phone Provisioning, Technical Bulletin 11785](#)

[Activating Feature Licensing on SoundPoint IP, SoundStation IP, and VVX Phones, Technical Bulletin 32265](#)

[Mutual Transport Layer Security Provisioning Using Microsoft Internet Information Services 6.0, Technical Bulletin 52609](#)

[Restarting and Resetting SoundPoint IP, SoundStation IP and Polycom VVX 1500 Phones, Quick Tip 18298](#)

[Register Standalone SoundPoint IP, SoundStation IP and Polycom VVX1500 Phones, Quick Tip 44011](#)

Trademarks

©2011, Polycom, Inc. All rights reserved.

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

Disclaimer

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

Limitation of Liability

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