CHAPTER 3

In-House Preprovisioning and Provisioning Servers

Cisco IP Telephony devices, other than RC units, are preprovisioned by the service provider with a profile. That pre-provision profile can range from a a limited set of parameters that resynchronizes the IP Telephony device to another profile with a complete set of parameters delivered by remote server. Or, it can be a complete set of parameters. By default, the IP Telephony device resynchronizes on power up and at intervals configured in the profile. When the user connects the IP Telephony device at the customer premises, the device downloads the updated profile and any firmware updates.

This process of preprovisioning, deployment, and remote provisioning can be accomplished many ways. This chapter describes the features and functionality available when preprovisioning the Cisco IP Telephony devices in-house and provisioning them remotely:

- Server Preparation and Software Tools, page 3-1
- In-House Device Preprovisioning, page 3-2
- Provisioning Server Setup, page 3-2

Server Preparation and Software Tools

The examples presented in this chapter require the availability of one or more servers. These servers can be installed and run on a local PC:

- TFTP (UDP port 69)
- syslog (UDP port 514)
- HTTP (TCP port 80)
- HTTPS (TCP port 443).

To troubleshoot server configuration, it is helpful to install clients for each type of server on a separate server machine. This establishes proper server operation, independent of the interaction with the Cisco IP Telephony devices.

Cisco also recommends the installation of the following software tools:

- To generate configuration profiles, it is useful to install the open source gzip compression utility.
- For profile encryption and HTTPS operations, install the open source OpenSSL software package.

- To test the dynamic generation of profiles and one-step remote provisioning using HTTPS, a scripting language with CGI scripting support, such as open source Perl language tools, is recommended.
- To verify secure exchanges between provisioning servers and the Cisco IP Telephony devices, install an Ethernet packet sniffer (such as the freely downloadable Ethereal/Wireshark). Capture an Ethernet packet trace of the interaction between the IP Telephony device and the provisioning server by running the packet sniffer on a PC that is connected to a switch with port mirroring enabled. For HTTPS transactions, you can use the ssldump utility.

In-House Device Preprovisioning

With the Cisco factory default configuration, an IP Telephony device automatically tries to resync to a profile on a TFTP server. The information regarding the profile and TFTP server configured for preprovisioning is delivered to the device by a managed DHCP server on a LAN. The service provider connects each new IP Telephony device that LAN and the IP Telephony device automatically resyncs to the local TFTP server, initializing its internal state in preparation for deployment. This preprovisioning profile typically includes the URL of a remote provisioning server that will keep the device updated after it is deployed and connected to the customer network.

The preprovisioned device bar code can be scanned to record its MAC address or serial number before the IP Telephony device is shipped to the customer. This information can be used to create the profile to which the IP Telephony device will resynchronize.

Upon receiving the IP Telephony device, the customer connects it to the broadband link. On power-up the IP Telephony device contacts the provisioning server through the URL configured through preprovisioning to for its resync and updates the profile and firmware as necessary.

Provisioning Server Setup

This section describes setup requirements for provisioning an IP Telephony device by using various servers and different scenarios. For testing purposes and for the purposes of this document, provisioning servers are installed and run on a local PC. Also, generally available software tools are useful for provisioning the Cisco IP Telephony devices.

TFTP Provisioning

Cisco IP Telephony devices support TFTP for both provisioning resync and firmware upgrade operations. When devices are deployed remotely, HTTP is recommended for provisioning as it offers greater reliability, given NAT and router protection mechanisms. TFTP is useful for the in-house preprovisioning of a large number of unprovisioned devices. See the "In-House Device Preprovisioning" section on page 3-2 for more information.

The IP Telephony device is able to obtain a TFTP server IP address directly from the DHCP server through DHCP option 66. If a Profile_Rule is configured with the filepath of that TFTP server, the device downloads its profile from the TFTP server when it is connected to a LAN and powered up.

The Profile_Rule provided with the factory default configuration is /*device*.cfg. For example, on a CP-8831-3PCC the filename is CP-8831-3PCC.cfg. If the device has the factory default profile, when powered up it resyncs to this file on the local TFTP server specified by DHCP option 66. (The filepath is relative to the TFTP server virtual root directory.)

Remote Endpoint Control and NAT

The IP Telephony device accesses the Internet through a router by using network address translation (NAT). For enhanced security, the router might attempt to block unauthorized incoming packets by implementing symmetric NAT (a packet filtering strategy that severely restricts the packets that are allowed to enter the protected network from the Internet). For this reason, remote provisioning by using TFTP is not recommended.

Voice over IP can co-exist with NAT only when some form of NAT traversal is provided. Configure Simple Traversal of UDP through NAT (STUN). This option requires that the user have (1) a dynamic external (public) IP address from your service, (2) a computer running STUN server software, and (3) an edge device with an asymmetric NAT mechanism.

HTTP Provisioning

The IP Telephony device behaves like a browser requesting web pages from a remote Internet site. This provides a reliable means of reaching the provisioning server, even when a customer router implements symmetric NAT or other protection mechanisms. HTTP and HTTPS work more reliably than TFTP in remote deployments, especially when the deployed units are connected behind residential firewalls or NAT-enabled routers.

Basic HTTP-based provisioning relies on the HTTP GET method for retrieving configuration profiles. Typically, a configuration file is created for each deployed IP Telephony device, and these files are stored within a HTTP server directory. When the server receives the GET request, it simply returns the file specified in the GET request header.

Alternatively, the requested URL can invoke a CGI script (using the GET method). The configuration profile is generated dynamically by querying a customer database and producing the profile on-the-fly.

When CGI handles resync requests, the IP Telephony device can use the HTTP POST method to request the resync configuration data. The device can be configured to convey certain status and identification information to the server within the body of the HTTP POST request. The server uses this information to generate a desired response configuration profile, or store the status information for later analysis and tracking.

As part of both GET and POST requests, the IP Telephony device automatically includes basic identifying information in the request header, in the User-Agent field. This information conveys the manufacturer, product name, current firmware version, and product serial number of the device.

For example, the following example is the User-Agent request field from a CP-8831-3PCC :

```
User-Agent: cisco/CP-8831-3PCC (88012BA01234)
```

When the IP Telephony device is configured to resync to a configuration profile by using HTTP, it is recommended that the profile be encrypted to protect confidential information. The IP Telephony device supports 256-bit AES in CBC mode to decrypt profiles. Encrypted profiles downloaded by the IP Telephony device by using HTTP avoid the danger of exposing confidential information contained in the configuration profile. This resync mode produces a lower computational load on the provisioning server when compared to using HTTPS.

Note The Cisco Unified IP Conference Phone 8831 for Third-Party Call Control supports HTTP Version 1.0, HTTP Version1.1, and Chunk Encoding when HTTP Version 1.1 is the negotiated transport protocol.

HTTP Status Code Handling on Resync and Upgrade

The phone supports improved HTTP response for remote provisioning (Resync). Current phone behavior is categorized in 3 ways:

- A—Success, where the subsequent requests are determined by "Resync Periodic" and "Resync Random Delay" values.
- B—Failure when File Not Found or corrupt profile. Subsequent requests are determined by "Resync Error Retry Delay" value.
- C—Other failure when there is a connection error caused by bad URL or IP address. Subsequent requests are determined by "Resync Error Retry Delay" value. When the request times out:”

Table 3-1 Phone Behavior for HTTP Responses

HTTP Status Code	Description	Phone Behavior
301 Moved Permanently	This and future requests should be directed to a new location.	Retry request immediately with new location.
302 Found	Known as Temporarily Moved.	Retry request immediately with new location.
3xx	Other 3xx responses not processed.	C
400 Bad Request	The request cannot be fulfilled due to bad syntax.	C
401 Unauthorized	Basic or digest access authentication challenge.	Immediate retry request with authentication credentials. Maximum 2 retries. Upon failure, the phone’s behavior is C.
403 Forbidden	Server refuses to respond.	C
404 Not Found	Requested resource not found. Subsequent requests by client are permissible.	B
407 Proxy Authentication Required	Basic or digest access authentication challenge.	Immediate retry request with authentication credentials. Maximum 2 retries. Upon failure, the phone’s behavior is C.
4xx	Other client error status codes are not processed.	C
500 Internal Server Error	Generic error message.	The IP conference phone 8831 behavior is C.
501 Not Implemented	The server does not recognize the request method, or it lacks the ability to fulfill the request.	The IP conference phone 8831 behavior is C.
502 Bad Gateway	The server is acting as a gateway or proxy and receives an invalid response from the upstream server.	The IP Conference Phone 8831 behavior is C.
503 Service Unavailable	The server is currently unavailable (overloaded or down for maintenance). This is a temporary state.	The IP Conference Phone 8831 behavior is C.

Table 3-1 Phone Behavior for HTTP Responses (continued)

HTTP Status Code	Description	Phone Behavior
504 Gateway Timeout	The server behaves as a gateway or proxy and does not receive timely response from the upstream server.	C
5xx	Other server error	C

HTTPS Provisioning

For increased security managing remotely deployed units, the IP Telephony device supports HTTPS for provisioning. Each IP Telephony device carries a unique SLL Client Certificate (and associated private key), in addition to a Sipura CA server root certificate. The latter allows the IP Telephony device to recognize authorized provisioning servers, and reject non-authorized servers. On the other hand, the client certificate allows the provisioning server to identify the individual device that issues the request.

For a service provider to manage deployment by using HTTPS, a server certificate must be generated for each provisioning server to which an IP Telephony device resyncs by using HTTPS. The server certificate must be signed by the Cisco Server CA Root Key, whose certificate is carried by all deployed units. To obtain a signed server certificate, the service provider must forward a certificate signing request to Cisco, which signs and returns the server certificate for installation on the provisioning server.

The provisioning server certificate must contain the Common Name (CN) field, and the FQDN of the host running the server in the subject. It might optionally contain information following the host FQDN, separated by a slash (/) character. The following examples are of CN entries that are accepted as valid by the IP Telephony device:

```
CN=sprov.callme.com
CN=pv.telco.net/mailto:admin@telco.net
CN=prof.voice.com/info@voice.com
```

In addition to verifying the server certificate, the IP Telephony device tests the server IP address against a DNS lookup of the server name specified in the server certificate.

A certificate signing request can be generated by using the OpenSSL utility. The following example shows the **openssl** command that produces a 1024-bit RSA public/private key pair and a certificate signing request:

```
openssl req –new –out provserver.csr
```

This command generates the server private key in **privkey.pem** and a corresponding certificate signing request in **provserver.csr**. The service provider keeps the **privkey.pem** secret and submits **provserver.csr** to Cisco for signing. Upon receiving the **provserver.csr** file Cisco generates **provserver.crt**, the signed server certificate.

Cisco also provides a Sipura CA Client Root Certificate to the service provider. This root certificate certifies the authenticity of the client certificate carried by each IP Telephony device. The Cisco Unified IP Conference Phone 8831 for Third-Party Call Control also supports third-party signed certificates such as those provided by Verisign, Cybertrust, and so forth.

The unique client certificate offered by each device during an HTTPS session carries identifying information embedded in its subject field. This information can be made available by the HTTPS server to a CGI script invoked to handle secure requests. In particular, the certificate subject indicates the unit product name (OU element), MAC address (S element), and serial number (L element). The following example from the Cisco Unified IP Conference Phone 8831 for Third-Party Call Control client certificate subject field shows these elements:

```
OU=CP-8831-3PCC, L=88012BA01234, S=000e08abcdef
```

Units manufactured before firmware 2.0.x do not contain individual SSL client certificates. When these units are upgraded to a firmware release in the 2.0.x tree, they become capable of connecting to a secure server using HTTPS, but are only able to supply a generic client certificate if requested to do so by the server. This generic certificate contains the following information in the identifying fields:

```
OU=cisco.com, L=ciscogeneric, S=ciscogeneric
```

To determine if an IP Telephony device carries an individualized certificate, use the $CCERT provisioning macro variable. The variable value expands to either Installed or Not Installed, according to the presence or absence of a unique client certificate. In the case of a generic certificate, it is possible to obtain the serial number of the unit from the HTTP request header in the User-Agent field.

HTTPS servers can be configured to request SSL certificates from connecting clients. If enabled, the server can verify the client certificate by using the Sipura CA Client Root Certificate supplied by Cisco. It can then provide the certificate information to a CGI for further processing.

The location for storing certificates might vary. For example, on an Apache installation the file paths for storing the provisioning server–signed certificate, its associated private key, and the Sipura CA client root certificate are as follows:

```
# Server Certificate:
SSLCertificateFile /etc/httpd/conf/provserver.crt

# Server Private Key:
SSLCertificateKeyFile /etc/httpd/conf/provserver.key

# Certificate Authority (CA):
SSLCACertificateFile /etc/httpd/conf/spacroot.crt
```

Refer to the documentation provided for a HTTPS server for specific information.

Firmware release 2.0.6 and higher supports the following cipher suites for SSL connection to a server by using HTTPS.

Table 3-2 Cipher Suites Supported for Connecting to an HTTPS Server

Numeric Code	Cipher Suite
0x0039	TLS_DHE_RSA_WITH_AES_256_CBC_SHA
0x0035	TLS_RSA_WITH_AES_256_CBC_SHA
0x0033	TLS_DHE_RSA_WITH_AES_128_CBC_SHA
0x002f	TLS_RSA_WITH_AES_128_CBC_SHA
0x0005	TLS_RSA_WITH_RC4_128_SHA
0x0004	TLS_RSA_WITH_RC4_128_MD5
0x0062	TLS_RSA_EXPORT1024_WITH_RC4_56_SHA
0x0060	TLS_RSA_EXPORT1024_WITH_RC4_56_MD5
0x0003	TLS_RSA_EXPORT_WITH_RC4_40_MD5

Redundant Provisioning Servers

The provisioning server can be specified as an IP address or as a fully qualified domain name (FQDN). The use of a FQDN facilitates the deployment of redundant provisioning servers. When the provisioning server is identified through a FQDN, the IP Telephony device attempts to resolve the FQDN to an IP address through DNS. Only DNS A-records are supported for provisioning; DNS SRV address resolution is not available for provisioning. The IP Telephony device continues to process A-records until a server responds. If no server associated with the A-records responds, the IP Telephony device logs an error to the syslog server.

Syslog Server

If a syslog server is configured on the IP Telephony device (using the <Syslog_Server> parameters), the resync and upgrade operations log messages to the syslog server. A message can be generated at the start of a remote file request (configuration profile or firmware load), and at the conclusion of the operation (indicating either success or failure).

The logged messages are configured in the following parameters and macro expanded into the actual syslog messages:

- Log_Request_Msg
- Log_Success_Msg
- Log_Failure_Msg

The Cisco Client Certificate Root Authority signs each unique certificate. The corresponding root certificate is made available to service providers for client authentication purposes.