

Security

This chapter describes the phone authentication support in Cisco Unified Communications Manager Express (Cisco Unified CME), Hypertext Transfer Protocol Secure (HTTPS) provisioning for Cisco Unified IP Phones, and the Media Encryption (SRTP) on Cisco Unified CME feature that provides the following secure voice call capabilities:

- Secure call control signaling and media streams in Cisco Unified CME networks using Secure Real-Time Transport Protocol (SRTP) and H.323 protocols.
- Secure supplementary services for Cisco Unified CME networks using H.323 trunks.
- Secure Cisco VG224 Analog Phone Gateway endpoints.

- Prerequisites for Security, on page 1
- Restrictions for Security, on page 2
- Information About Security, on page 2
- Configure Security, on page 19
- Configuration Examples for Security, on page 64
- Where to Go Next, on page 80
- Feature Information for Security, on page 81

Prerequisites for Security

- Cisco Unified CME 4.0 or a later version for Phone Authentication.
- Cisco Unified CME 4.2 or a later version for Media Encryption (SRTP) on Cisco Unified CME.
- Cisco IOS feature set Advanced Enterprise Services (adventerprisek9) or Advanced IP Services (advipservicesk9) on supported platforms.
- Firmware 9.0(4) or a later version must be installed on the IP phone for HTTPS provisioning.
- System clock must be set by using one of the following methods:
  - Configure Network Time Protocol (NTP). For configuration information, see Enable Network Time Protocol.
  - Manually set the software clock using the **clock set** command. For information about this command, see Cisco IOS Network Management Command Reference.

Restrictions for Security

Phone Authentication

- Cisco Unified CME phone authentication is not supported on the Cisco IAD 2400 series or the Cisco 1700 series.

Media Encryption

- Secure three-way software conferencing is not supported. A secure call beginning with SRTP will always fall back to nonsecure Real-Time Transport Protocol (RTP) when it is joined to a conference.
- If a party drops from a three-party conference, the call between the remaining two parties returns to secure if the two parties are SRTP-capable local Skinny Client Control Protocol (SCCP) endpoints to a single Cisco Unified CME and the conference creator is one of the remaining parties. If either of the two remaining parties are only RTP-capable, the call remains nonsecure. If the two remaining parties are connected through FXS, PSTN, or VoIP, the call remains nonsecure.
- Calls to Cisco Unity Connection are not secure.
- Music On Hold (MOH) is not secure.
- Video calls are not secure.
- Modem relay and T.3 fax relay calls are not secure.
- Media flow-around is not supported for call transfer and call forward.
- Conversion between inband tone and RFC 2833 DTMF is not supported. RFC 2833 DTMF handling is supported when encryption keys are sent to secure DSP Farm devices but is not supported for codec passthrough.
- Secure Cisco Unified CME supports SIP trunks and H.323 trunks only on the Cisco Integrated Services Router Generation 2 platform. Secure Unified CME is not supported on Cisco 4000 Series Integrated Services Routers.
- Secure calls are supported in the default session application only.

Information About Security

Unified CME Password Policy

From Unified CME 12.6 Release (Cisco IOS XE Gibraltar 16.11.1a) onwards, all configurations on Unified CME must meet the Unified CME password policy.

General password policy guidelines:

- Passwords must have a minimum of 6 alphanumeric characters, and a maximum of 15 alphanumeric characters.
- Passwords must not contain symbols or special characters.

- Passwords must contain at least one numeral, one uppercase alphabet, and one lowercase alphabet.

If the password is not configured as per the policy, the Unified CME router displays an error message:

```
Error: The password you have entered is incorrect.
Your password must contain:
1. A minimum of 6 and a maximum of 15 alphanumeric characters, excluding symbols and special
 characters.
2. A minimum of one numeral, one uppercase alphabet, and one lowercase alphabet.
```

Note The Unified CME password policy is applicable for Unified CME configurations on Cisco IOS XE 16.11.1a and later.

Unified CME password policy is not applicable in the following scenarios:

- Upgrade from an older IOS version to Cisco IOS XE 16.11.1a
- Downgrade from Cisco IOS XE 16.11.1a to an older version.

Guidelines for Password Configuration and Encryption

Configure the passwords relevant to Unified CME using the CLI commands as follows:

- **voice reg pool** configuration mode
  - **username** *name* **password [0|6]** *password*
  - **ata-ivr-pwd [0|6]** *password*
- **voice register global** (for auto register) configuration mode
  - **password [0|6]** *password*
- **ephone** configuration mode
  - **username** *name* **password [0|6]** *password*
- **telephony-service** configuration mode
  - **ssh userid** *user-id-name* **password [0|6]** *password*
  - **service local-directory authenticate** *username* **[0|6]** *password*
  - **xml user** *username* **password [0|6]** *password privilege-level*
  - **standby user** *username* **password [0|6]** *password*
- Extension Mobility Related (under **telephony-service** configuration mode) configuration mode
  - **url authentication** *url-address application-name* **password [0|6]** *password*
  - **authentication credential** *application-name* **password [0|6]** *password*
- Extension Mobility Related (under **voice logout-profile** configuration mode) configuration mode

- **user** *name* **password [0|6]** *password*

- **voice user-profile** , **voice logout-profile** , and **voice reg pool** configuration mode
  - **pin [0|6]** *pin*

- **voice user-profile** configuration mode
  - **username** *name* **password [0|6]** *password*

The following are some of the configuration recommendations for Unified CME Password Policy:

- The **0** in the parameter **[0|6]** mentioned in the CLI command represents plain, unencrypted text and **6** represents level 6 password encryption.
- Apart from the parameter configurations (**[0|6]**) at the command level, the Unified CME router must be configured to support encryption. Configure the CLI command **encrypt password** to support type 6 encryption on the Unified CME router.
- The CLI command **encrypt password** is enabled by default on Unified CME router. However, you must mandatorily configure **key config-key password-encrypt** *[key]* and **password encryption aes** to support encryption on the Unified CME router. For a sample configuration, see Example for Configuring Unified CME for Password Policy , on page 65
- If the key used to encrypt the password is replaced with a new key (replace key or re-key), then the password is re-encrypted with the new key.
- You must adhere to CME Password Policy for both type 0 and type 6 parameters that you configure on Unified CME. For more information on CME Password Policy, see Unified CME Password Policy, on page 2.

Note For the CLI command **ata-ivr-pwd** , you need to use a four digit character string as password. For more information, see the CLI command **ata-ivr-pwd** in Unified CME Command Reference Guide.

The following table provides information on password encryption levels that are supported in Unified CME:

Table 1: Password Encryption Configuration

User Input	encrypt password + key config-key password-encrypt *[key]* + password encryption aes	Password Encryption Status
Encrypted text (Type 6)	• **encrypt password** —Enabled • **key config-key password-encrypt** *[key]*—Enabled • **password encryption aes**—Enabled	Encrypted

User Input	encrypt password + key config-key password-encrypt *[key]* + password encryption aes	Password Encryption Status
Encrypted text (Type 6)	• **encrypt password** —Disabled • **key config-key password-encrypt** *[key]*—Enabled • **password encryption aes**—Enabled	Unencrypted (Plain Text)
Plain text (Type 0)	• **encrypt password** —Disabled • **key config-key password-encrypt** *[key]*—Enabled • **password encryption aes**—Enabled	Unencrypted (Plain Text)
Plain text (Type 0)	• **encrypt password** —Enabled • **key config-key password-encrypt** *[key]*—Enabled • **password encryption aes**—Enabled	Encrypted

Note Configure the CLI command **no encrypt password** to disable password encryption.

Downgrade Consideration for Password Encryption

If you are performing a downgrade from Unified CME 12.6 to an earlier version, then you must execute the CLI command **no encrypt password** . If the CLI command **no encrypt password** is configured, the password is presented as plain text.

Removal of Passwords and Keys from Logs

From Unified CME Release 12.6 onwards, passwords and sRTP keys are not printed to logs to enhance security of Unified CME. The information about keys is available only in the show commands from Unified CME 12.6 release onwards. The CLI command **show ephone offhook** for SCCP and **show sip-ua calls** for SIP are enhanced to display the keys that are in use per media stream, along with the sRTP Ciphers.

For a sample output, see Example for Password and Key Removal from Logs, on page 64.

Deprecation of CLI Commands

From Unified CME Release 12.6 onwards, the following CLI commands that are configured under **telephony-service** configuration mode are deprecated to enhance product security:

- **log password** *password-string*
- **xmltest**
- **xmlschema** *schema-url*
- **xmlthread** *number*

For more information on the deprecated commands, see Cisco Unified Communications Manager Express Command Reference.

Phone Authentication Overview

Phone authentication is a security infrastructure for providing secure SCCP signaling between Cisco Unified CME and IP phones. The goal of Cisco Unified CME phone authentication is to create a secure environment for a Cisco Unified CME IP telephony system.

Phone authentication addresses the following security needs:

- Establishing the identity of each endpoint in the system
- Authenticating devices
- Providing signaling-session privacy
- Providing protection for configuration files

Cisco Unified CME phone authentication implements authentication and encryption to prevent identity theft of the phone or Cisco Unified CME system, data tampering, call-signaling tampering, or media-stream tampering. To prevent these threats, the Cisco Unified IP telephony network establishes and maintains authenticated communication streams, digitally signs files before they are transferred to phones, and encrypts call signaling between Cisco Unified IP phones.

Cisco Unified CME phone authentication depends on the following processes:

- Phone Authentication, on page 6
- File Authentication, on page 7
- Signaling Authentication, on page 7

Phone Authentication

The phone authentication process occurs between the Cisco Unified CME router and a supported device when each entity accepts the certificate of the other entity; only then does a secure connection between the entities occur. Phone authentication relies on the creation of a Certificate Trust List (CTL) file, which is a list of known, trusted certificates and tokens. Phones communicate with Cisco Unified CME using a Transport Layer Security (TLS) session connection, which requires that the following criteria be met:

- A certificate must exist on the phone.

- A phone configuration file must exist on the phone, and the Cisco Unified CME entry and certificate must exist in the file.

File Authentication

The file authentication process validates digitally signed files that a phone downloads from a Trivial File Transfer Protocol (TFTP) server—for example, configuration files, ring list files, locale files, and CTL files. When the phone receives these types of files from the TFTP server, the phone validates the file signatures to verify that file tampering did not occur after the files were created.

Signaling Authentication

The signaling authentication process, also known as signaling integrity, uses the TLS protocol to validate that signaling packets have not been tampered with during transmission. Signaling authentication relies on the creation of the CTL file.

Public Key Infrastructure

Cisco Unified CME phone authentication uses the public-key-infrastructure (PKI) capabilities in Cisco IOS software for certificate-based authentication of IP phones. PKI provides customers with a scalable, secure mechanism for distributing, managing, and revoking encryption and identity information in a secure data network. Every entity (a person or a device) participating in the secure communication is enrolled in the PKI using a process in which the entity generates a Rivest-Shamir-Adleman (RSA) key pair (one private key and one public key) and has its identity validated by a trusted entity (also known as a certification authority [CA] or trustpoint).

After each entity enrolls in a PKI, every peer (also known as an end host) in a PKI is granted a digital certificate that has been issued by a CA.

When peers must negotiate a secure communication session, they exchange digital certificates. Based on the information in the certificate, a peer can validate the identity of another peer and establish an encrypted session with the public keys contained in the certificate.

Phone Authentication Components

A variety of components work together to ensure secure communications in a Cisco Unified CME system. Table 2: Cisco Unified CME Phone Authentication Components , on page 7 describes the Cisco Unified CME phone authentication components.

Table 2: Cisco Unified CME Phone Authentication Components

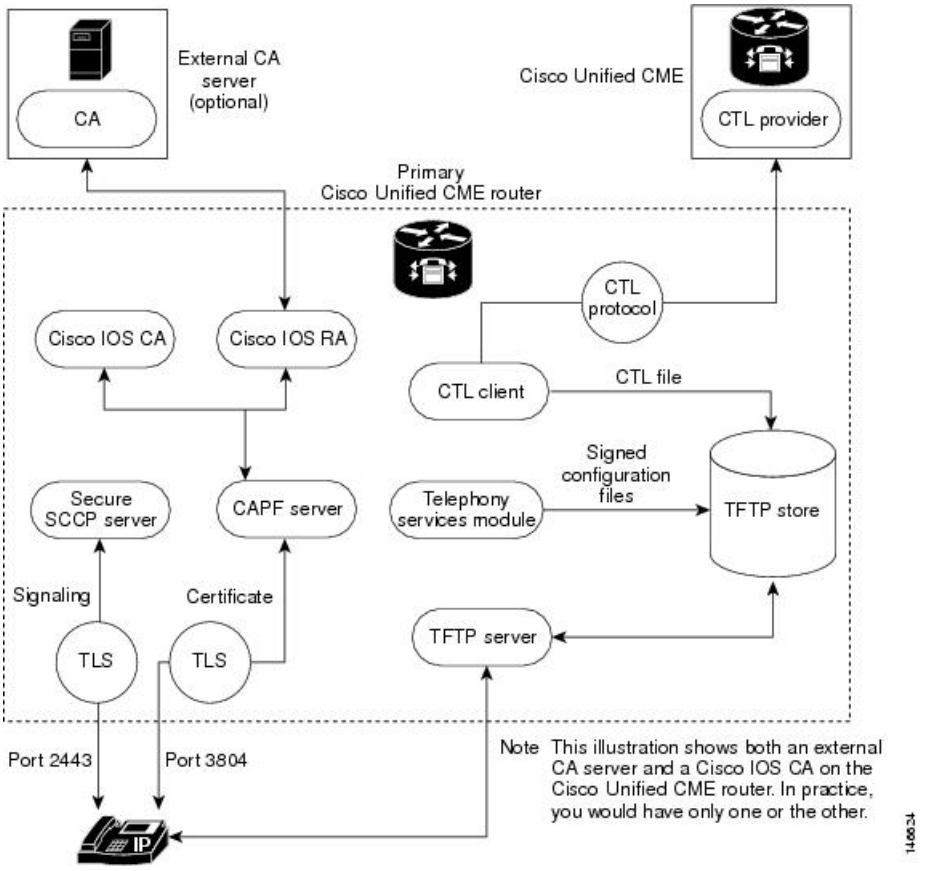
Component	Definition
certificate	An electronic document that binds a user's or device's name to its public key. Certificates are commonly used to validate digital signatures. Certificates are needed for authentication during secure communication. An entity obtains a certificate by enrolling with the CA.
signature	An assurance from an entity that the transaction it accompanies is authentic. The entity's private key is used to sign transactions and the corresponding public key is used for decryption.

Component	Definition
RSA key pair	RSA is a public key cryptographic system developed by Ron Rivest, Adi Shamir, and Leonard Adleman. An RSA key pair consists of a public key and a private key. The public key is included in a certificate so that peers can use it to encrypt data that is sent to the router. The private key is kept on the router and used both to decrypt the data sent by peers and to digitally sign transactions when negotiating with peers. You can configure multiple RSA key pairs to match policy requirements, such as key length, key lifetime, and type of keys, for different certificate authorities or for different certificates.
certificate server trustpoint	A certificate server generates and issues certificates on receipt of legitimate requests. A trustpoint with the same name as the certificate server stores the certificates. Each trustpoint has one certificate plus a copy of the CA certificate.
certification authority (CA)	The root certificate server. It is responsible for managing certificate requests and issuing certificates to participating network devices. This service provides centralized key management for participating devices and is explicitly trusted by the receiver to validate identities and to create digital certificates. The CA can be a Cisco IOS CA on the Cisco Unified CME router, a Cisco IOS CA on another router, or a third-party CA.
registration authority (RA)	Records or verifies some or all of the data required for the CA to issue certificates. It is required when the CA is a third-party CA or Cisco IOS CA is not on the Cisco Unified CME router.
certificate trust list (CTL) file CTL client CTL provider	A mandatory structure that contains the public key information (server identities) of all the servers with which the IP phone needs to interact (for example, the Cisco Unified CME server, TFTP server, and CAPF server). The CTL file is digitally signed by the SAST. After you configure the CTL client, it creates the CTL file and makes it available in the TFTP directory. The CTL file is signed using the SAST certificate's corresponding private key. An IP phone is then able to download this CTL file from the TFTP directory. The filename format for each phone's CTL file is CTLSEP<mac-addr>.tlv. When the CTL client is run on a router in the network that is not a Cisco Unified CME router, you must configure a CTL provider on each Cisco Unified CME router in the network. Similarly, if a CTL client is running on one of two Cisco Unified CME routers in a network, a CTL provider must be configured on the other Cisco Unified CME router. The CTL protocol transfers information to and from the CTL provider that allows the second Cisco Unified CME router to be trusted by phones and vice versa.
certificate revocation list (CRL)	File that contains certificate expiration dates and used to determine whether a certificate that is presented is valid or revoked.

Component	Definition
system administrator security token (SAST)	Part of the CTL client that is responsible for signing the CTL file. The Cisco Unified CME certificate and its associated key pair are used for the SAST function. There are actually two SAST records pertaining to two different certificates in the CTL file for security reasons. They are known as SAST1 and SAST2. If one of the certificates is lost or compromised, then the CTL client regenerates the CTL file using the other certificate. When a phone downloads the new CTL file, it verifies with only one of the two original public keys that was installed earlier. This mechanism is to prevent IP phones from accepting CTL files from unknown sources.
certificate authority proxy function (CAPF)	Entity that issues certificates (LSCs) to phones that request them. The CAPF is a proxy for the phones, which are unable to directly communicate with the CA. The CAPF can also perform the following certificate-management tasks: • Upgrade existing locally significant certificates on the phones. • Retrieve phone certificates for viewing and troubleshooting. • Delete LSCs on the phone.
manufacture-installed certificate (MIC) locally significant certificate (LSC)	Phones need certificates to engage in secure communications. Many phones come from the factory with MICs, but MICs may expire or become lost or compromised. Some phones do not come with MICs. LSCs are certificates that are issued locally to the phones using the CAPF server.
transport Layer Security (TLS) protocol	IETF standard (RFC 2246) protocol, based on Netscape Secure Socket Layer (SSL) protocol. TLS sessions are established using a handshake protocol to provide privacy and data integrity. The TLS record layer fragments and defragments, compresses and decompresses, and performs encryption and decryption of application data and other TLS information, including handshake messages.

Figure 1: Cisco Unified CME Phone Authentication, on page 10 shows the components in a Cisco Unified CME phone authentication environment.

Figure 1: Cisco Unified CME Phone Authentication



Phone Authentication Process

The following is a high-level summary of the phone-authentication process.

To enable Cisco Unified CME phone authentication:

1. Certificates are issued.

   The CA issues certificates to Cisco Unified CME, SAST, CAPF, and TFTP functions.

2. The CTL file is created, signed and published.

   a. The CTL file is created by the CTL client, which is configuration driven. Its goal is to create a CTLfile.tlv for each phone and deposit it in the TFTP directory. To complete its task, the CTL client needs the certificates and public key information of the CAPF server, Cisco Unified CME server, TFTP server, and SASTs.

   b. The CTL file is signed by the SAST credentials. There are two SAST records pertaining to two different certificates in the CTL file for security reasons. If one of the certificates is lost or compromised, then the CTL client regenerates the CTL file using the other certificate. When a phone downloads the new CTL file, it verifies the download with only one of the two original public keys that was installed earlier. This mechanism prevents IP phones from accepting CTL files from unknown sources.

c. The CTL file is published on the TFTP server. Because an external TFTP server is not supported in secure mode, the configuration files are generated by the Cisco Unified CME system itself and are digitally signed by the TFTP server's credentials. The TFTP server credentials can be the same as the Cisco Unified CME credentials. If desired, a separate certificate can be generated for the TFTP function if the appropriate trustpoint is configured under the CTL-client interface.

3. The telephony service module signs phone configuration files and each phone requests its file.
4. When an IP phone boots up, it requests the CTL file (CTLfile.tlv) from the TFTP server and downloads its digitally signed configuration file, which has the filename format of SEP<mac-address>.cnf.xml.sgn.
5. The phone then reads the CAPF configuration status from the configuration file. If a certificate operation is needed, the phone initiates a TLS session with the CAPF server on TCP port 3804 and begins the CAPF protocol dialogue. The certificate operation can be an upgrade, delete, or fetch operation. If an upgrade operation is needed, the CAPF server makes a request on behalf of the phone for a certificate from the CA. The CAPF server uses the CAPF protocol to obtain the information it needs from the phone, such as the public key and phone ID. After the phone successfully receives a certificate from the server, the phone stores it in its flash memory.
6. With the certificate in its flash, the phone initiates a TLS connection with the secure Cisco Unified CME server on a well-known TCP port (2443) if the device security mode settings in the .cnf.xml file are set to authenticated or encrypted. This TLS session is mutually authenticated by both parties. The IP phone knows the Cisco Unified CME server's certificate from the CTL file, which it initially downloaded from the TFTP server. The phone's LSC is a trusted party for the Cisco Unified CME server because the issuing CA certificate is present in the router.

Startup Messages

If the certificate server is part of your startup configuration, you may see the following messages during the boot procedure:

```
% Failed to find Certificate Server's trustpoint at startup
% Failed to find Certificate Server's cert.
```

These messages are informational messages that show a temporary inability to configure the certificate server because the startup configuration has not been fully parsed yet. The messages are useful for debugging if the startup configuration has been corrupted.

Configuration File Maintenance

In a secure environment, several types of configuration files must be digitally signed before they can be hosted and used. The filenames of all signed files have a .sgn suffix.

The Cisco Unified CME telephony service module creates phone configuration files (.cnf.xml suffix) and hosts them on a Cisco IOS TFTP server. These files are signed by the TFTP server's credentials.

In addition to the phone configuration files, other Cisco Unified CME configuration files such as the network and user-locale files must be signed. These files are internally generated by Cisco Unified CME, and the signed versions are automatically created in the current code path whenever the unsigned versions are updated or created.

Other configuration files that are not generated by Cisco Unified CME, such as ringlist.xml, distinctiveringlist.xml, audio files, and so forth, are often used for Cisco Unified CME features. Signed versions of these configuration files are not automatically created. Whenever a new configuration file that has not been generated by Cisco Unified CME is imported into Cisco Unified CME, use the **load-cfg-file** command, which does all of the following:

- Hosts the unsigned version of the file on the TFTP server.
- Creates a signed version of the file.
- Hosts the signed version of the file on the TFTP server.

You can also use the **load-cfg-file** command instead of the **tftp-server** command when only the unsigned version of a file needs to be hosted on the TFTP server.

CTL File Maintenance

The CTL file contains the SAST records and other records. (A maximum of two SAST records may exist.) The CTL file is digitally signed by one of the SAST credentials that are listed in the CTL file before the CTL file is downloaded by the phone and saved in its flash. After receiving the CTL file, a phone trusts a newer or changed CTL file only if it is signed by one of the SAST credentials that is present in the original CTL file.

For this reason, you should take care to regenerate the CTL file only with one of the original SAST credentials. If both SAST credentials are compromised and a CTL file must be generated with a new credential, you must reset the phone to its factory defaults.

CTL Client and Provider

The CTL client generates the CTL file. The CTL client must be provided with the names of the trustpoints it needs for the CTL file. It can run on the same router as Cisco Unified CME or on another, standalone router. When the CTL client runs on a standalone router (not a Cisco Unified CME router), you must configure a CTL provider on each Cisco Unified CME router. The CTL provider securely communicates the credentials of the Cisco Unified CME server functions to the CTL client that is running on another router.

When the CTL client is running on either a primary or secondary Cisco Unified CME router, you must configure a CTL provider on each Cisco Unified CME router on which the CTL client is not running.

The CTL protocol is used to communicate between the CTL client and a CTL provider. Using the CTL protocol ensures that the credentials of all Cisco Unified CME routers are present in the CTL file and that all Cisco Unified CME routers have access to the phone certificates that were issued by the CA. Both elements are prerequisites to secure communications.

To enable CTL clients and providers, see Configure the CTL Client, on page 29 and Configure the CTL Provider, on page 41.

Manually Importing MIC Root Certificate

When a phone uses a MIC for authentication during the TLS handshake with the CAPF server, the CAPF server must have a copy of the MIC to verify it. Different certificates are used for different types of IP phones.

A phone uses a MIC for authentication when it has a MIC but no LSC. For example, you have a Cisco Unified IP Phone 7970 that has a MIC by default but no LSC. When you schedule a certificate upgrade with the authentication mode set to MIC for this phone, the phone presents its MIC to the Cisco Unified CME CAPF

server for authentication. The CAPF server must have a copy of the MIC's root certificate to verify the phone's MIC. Without this copy, the CAPF upgrade operation fails.

To ensure that the CAPF server has copies of the MICs it needs, you must manually import certificates to the CAPF server. The number of certificates that you must import depends on your network configuration. Manual enrollment refers to copy-and-paste or TFTP transfer methods.

To manually import the MIC root certificate, see Manually Import the MIC Root Certificate, on page 48.

Feature Design of Media Encryption

Companion voice security Cisco IOS features provide an overall architecture for secure end-to-end IP telephony calls on supported network devices that enable the following:

- SRTP-capable Cisco Unified CME networks with secure interoperability
- Secure Cisco IP phone calls
- Secure Cisco VG224 Analog Phone Gateway endpoints
- Secure supplementary services

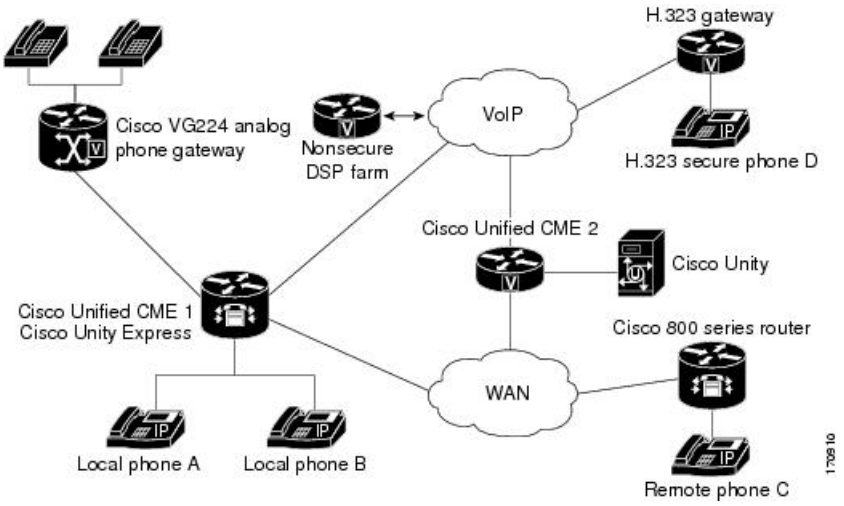
These features are implemented using media and signaling authentication and encryption in Cisco IOS H.323 networks. H.323, the ITU-T standard that describes packet-based video, audio, and data conferencing, refers to a set of other standards, including H.450, to describe its actual protocols. H.323 allows dissimilar communication devices to communicate with each other by using a standard communication protocol and defines a common set of codecs, call setup and negotiating procedures, and basic data transport methods. H.450, a component of the H.323 standard, defines signaling and procedures that are used to provide telephony-like supplementary services. H.450 messages are used in H.323 networks to implement secure supplementary service support and also empty capability set (ECS) messaging for media capability negotiation.

Secure Cisco Unified CME

The secure Cisco Unified CME solution includes secure-capable voice ports, SCCP endpoints, and a secure H.323 or SIP trunk between Cisco Unified CME and Cisco Unified Communications Manager for audio media. Figure 2: Secure Cisco Unified CME System, on page 14 shows the components of a secure Cisco Unified CME system.

Note Secure Unified CME is not supported on Cisco 4000 Series Integrated Services Routers.

Figure 2: Secure Cisco Unified CME System



Secure Cisco Unified CME implements call control signaling using Transport Layer Security (TLS) or IPsec (IP Security) for the secure channel and uses SRTP for media encryption. Secure Cisco Unified CME manages the SRTP keys to endpoints and gateways.

The Media Encryption (SRTP) on Cisco Unified CME feature supports the following features:

- SCCP endpoints.
- Secure voice calls in a mixed shared line environment that allows both RTP- and SRTP-capable endpoints; shared line media security depends on the endpoint configuration.
- Secure supplementary services using H.450 including:
  - Call forward
  - Call transfer
  - Call hold and resume
  - Call park and call pickup
  - Nonsecure software conference

Note SRTP conference calls over H.323 may experience a zero- to two-second noise interval when the call is joined to the conference.

- Secure calls in a non-H.450 environment.
- Secure Cisco Unified CME interaction with secure Cisco Unity.
- Secure Cisco Unified CME interaction with Cisco Unity Express (interaction is supported and calls are downgraded to nonsecure mode).
- Secure transcoding for remote phones with DSP Farm transcoding configured.

These features are discussed in the following sections.

Secure Supplementary Services

The Media Encryption (SRTP) feature supports secure supplementary services in both H.450 and non-H.450 Cisco Unified CME networks. A secure Cisco Unified CME network should be either H.450 or non-H.450, not a hybrid.

Secure SIP Trunk Support on Cisco Unified CME

Prior to Cisco Unified CME Relese 10 release, supplementary services were not supported on the secure SIP trunk of the secure SCCP Cisco Unified CME. This feature supports the following supplementary services in the secure SRTP and SRTP fallback modes on the SIP trunk of the SCCP Cisco Unified CME:

- Basic secure calls
- Call hold and resume
- Call transfer (blind and consult)
- Call forward (CFA,CFB,CFNA)
- DTMF support
- Call park and pickup
- Voice mail systems using CUE (works only with SRTP fallback mode)

To enable the supplementary services, use the existing "**supplementary-service media-renegotiate**" command as shown in the following example:

```
(config)# voice service voip
(conf-voi-serv)# no ip address trusted authenticate
(conf-voi-serv)# srtp
(conf-voi-serv)# allow-connections sip to sip
(conf-voi-serv)# no supplementary-service sip refer
(conf-voi-serv)# supplementary-service media-renegotiate
```



Note In the SRTP mode, nonsecure media (RTP) format is not allowed across the secure SIP trunk. For Music On Hold, Tone On Hold, and Ring Back Tone, the tone is not played across the SIP trunk. In SRTP fallback mode, media across the secure SIP trunk is switched over to RTP if the remote end is nonsecure or while playing the MMusic On Hold, Tone On Hold, and Ring Back Tone.

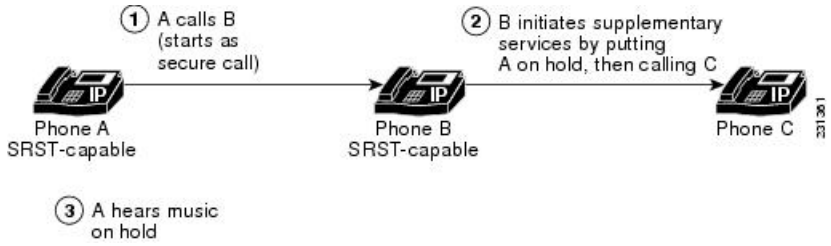
Restriction

- Secure SIP trunk is supported only on SCCP Cisco Unified CME and not on SIP Cisco Unified CME. Secure SIP lines are not supported on the Cisco Unified CME mode.
- Secure Unified CME is not supported on Cisco 4000 Series Integrated Services Routers.
- Xcoder support is not available for playing secure tones (Music On Hold, Tone On Hold, and Ring Back Tone).
- Tones are not played in the SRTP mode because these tones are available only in non-secure (RTP) format.
- We recommend that you configure **no supplementary-service sip refer** command for SCCP Cisco Unfied CME for the supplementary services.

Secure Cisco Unified CME in an H.450 Environment

Signaling and media encryption among secure endpoints is supported, enabling supplementary services such as call transfer (H.450.2) and call forward (H.450.3) between secure endpoints. Call park and pick up use H.450 messages. Secure Cisco Unified CME is H.450-enabled by default; however, secure music on hold (MOH) and secure conferences (three-way calling) are not supported. For example, when supplementary services are initiated as shown in Figure 3: Music on Hold in an H.450 Environment, on page 16, ECS and Terminal Capabilities Set (TCS) are used to negotiate the initially secure call between A and B down to RTP so A can hear MOH. When B resumes the call to A, the call goes back to SRTP. Similarly, when a transfer is initiated, the party being transferred is put on hold and the call is negotiated down to RTP. When the call is transferred, it goes back to SRTP if the other end is SRTP capable.

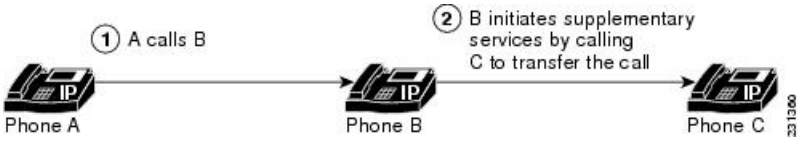
Figure 3: Music on Hold in an H.450 Environment



Secure Cisco Unified CME in a Non H.450 Environment

Security for supplementary services requires midcall key negotiation or midcall media renegotiation. In an H.323 network where there are no H.450 messages, media renegotiation is implemented using ECS for scenarios such as mismatched codecs and secure calls. If you disable H.450 on the router globally, the configuration is applied to RTP and SRTP calls. The signaling path is hairpin on XOR for Cisco Unified CME and Cisco Unified Communications Manager. For example, in Figure 4: Transfer in a Non-H.450 Environment, on page 17, the signaling path goes from A through B (the supplementary services initiator) to C. When deploying voice security in this scenario, consider that the media security keys will pass through XOR, that is, through B, the endpoint that issued the transfer request. To avoid the man-in-the-middle attack, the XOR must be a trusted entity.

Figure 4: Transfer in a Non-H.450 Environment



The media path is optional. The default media path for Cisco Unified CME is hairpin. However, whenever possible media flow around can be configured on Cisco Unified CME. When configuring media flow through, which is the default, remember that chaining multiple XOR gateways in the media path introduces more delay and thus reduces voice quality. Router resources and voice quality limit the number of XOR gateways that can be chained. The requirement is platform dependent and may vary between signaling and media. The practical chaining level is three.

A transcoder is inserted when there is a codec mismatch and ECS and TCS negotiation fails. For example, if Phone A and Phone B are SRTP capable, but Phone A uses the G.711 codec and Phone B uses the G.729 codec, a transcoder is inserted if Phone B has one. However, the call is negotiated down to RTP to fulfill the codec requirement so the call is not secure.

Secure Transcoding for Remote Phones with DSP Farm Transcoding Configured

Transcoding is supported for remote phones that have the **dspfarm-assist** keyword of the **codec** command configured. A remote phone is a phone that is registered to a Cisco Unified CME and is residing on a remote location across the WAN. To save bandwidth across the WAN connection, calls to such a phone can be made to use the G.729r8 codec by configuring the **codec g729r8 dspfarm assist** command for the ephone. The **g729r8** keyword forces calls to such a phone to use the G.729 codec. The **dspfarm-assist** keyword enables using available DSP resources if an H.323 call to the phone needs to be transcoded.

Note Transcoding is enabled only if an H.323 call with a different codec from the remote phone tries to make a call to the remote phone. If a local phone on the same Cisco Unified CME as the remote phone makes a call to the remote phone, the local phone is forced to change its codec to G.729 instead of using transcoding.

Secure transcoding for point-to-point SRTP calls can only occur when both the SCCP phone that is to be serviced by Cisco Unified CME transcoding and its peer in the call are SRTP capable and have successfully negotiated the SRTP keys. Secure transcoding for point-to-point SRTP calls cannot occur when only one of the peers in the call is SRTP capable.

If Cisco Unified CME transcoding is to be performed on a secure call, the Media Encryption (SRTP) on Cisco Unified CME feature allows Cisco Unified CME to provide the DSP Farm with the encryption keys for the secure call as additional parameters so that Cisco Unified CME transcoding can be performed successfully. Without the encryption keys, the DSP Farm would not be able to read the encrypted voice data to transcode it.

Note The secure transcoding described here does not apply to IP-IP gateway transcoding.

Cisco Unified CME transcoding is different from IP-to-IP gateway transcoding because it is invoked for an SCCP endpoint only, instead of for bridging VoIP call legs. Cisco Unified CME transcoding and IP-to-IP gateway transcoding are mutually exclusive, that is, only one type of transcoding can be invoked for a call.

If no DSP Farm capable of SRTP transcoding is available, Cisco Unified CME secure transcoding is not performed and the call goes through using G.711.

For configuration information, see Register the DSP Farm with Cisco Unified CME 4.2 or a Later Version in Secure Mode.

Secure Cisco Unified CME with Cisco Unity Express

Cisco Unity Express does not support secure signaling and media encryption. Secure Cisco Unified CME interoperates with Cisco Unity Express but calls between Cisco Unified CME and Cisco Unity Express are not secure.

In a typical Cisco Unity Express deployment with Cisco Unified CME in a secure H.323 network, Session Initiation Protocol (SIP) is used for signaling and the media path is G.711 with RTP. For Call Forward No Answer (CFNA) and Call Forward All (CFA), before the media path is established, signaling messages are sent to negotiate an RTP media path. If codec negotiation fails, a transcoder is inserted. The Media Encryption (SRTP) on Cisco Unified CME feature's H.323 service provider interface (SPI) supports fast start calls. In general, calls transferred or forwarded back to Cisco Unified CME from Cisco Unity Express fall into existing call flows and are treated as regular SIP and RTP calls.

The Media Encryption (SRTP) on Cisco Unified CME feature supports blind transfer back to Cisco Unified CME only. When midcall media renegotiation is configured, the secure capability for the endpoint is renegotiated regardless of which transfer mechanism, H.450.2 or Empty Capability Set (ECS), is used.

Secure Cisco Unified CME with Cisco Unity

The Media Encryption (SRTP) on Cisco Unified CME feature supports Cisco Unity 4.2 or a later version and Cisco Unity Connection 1.1 or a later version using SCCP. Secure Cisco Unity for Cisco Unified CME acts like a secure SCCP phone. Some provisioning is required before secure signaling can be established. Cisco Unity receives Cisco Unified CME device certificates from the Certificate Trust List (CTL) and Cisco Unity certificates are inserted into Cisco Unified CME manually. Cisco Unity with SIP is not supported.

The certificate for the Cisco Unity Connection is in the Cisco Unity administration web application under the "port group settings."

HTTPS Provisioning For Cisco Unified IP Phones

This section contains the following topics:

- HTTPS support for an External Server, on page 18
- HTTPS Support in Cisco Unified CME, on page 19

HTTPS support for an External Server

There is an increasing need to securely access web content on Cisco Unified IP phones using HTTPS. The X.509 certificate of a third-party web server must be stored in the IP phone's CTL file to authenticate the web server but the **server** command used to enter trustpoint information cannot be used to import the certificate to the CTL file. Because the **server** command requires the private key from the third-party web server for certificate chain validation and you cannot obtain that private key from the web server, the **import certificate** command is added to save the trusted certificate in the CTL file.

For information on how to import a trusted certificate to an IP phone's CTL file for HTTPS provisioning, see HTTPS Provisioning for Cisco Unified IP Phones, on page 58.

For information on phone authentication support in Cisco Unified CME, see Phone Authentication Overview, on page 6.

HTTPS Support in Cisco Unified CME

Cisco Unified IP phones use HTTP for some of the services offered by Cisco Unified CME. These services, which include local-directory lookup on Cisco Unified CME, My Phone Apps, and Extension Mobility, are invoked by pressing the "Services" button on the phones.

With Hypertext Transfer Protocol Secure (HTTPS) support in Cisco Unified CME 9.5 and later versions, these services can be invoked using an HTTPS connection from the phones to Cisco Unified CME.

Note Ensure that the configured phone is provisioned for HTTPS-based services that run on Cisco Unified CME before configuring HTTPS globally or locally. Please refer to the appropriate phone administrator guide to know if your Cisco Unified IP phone supports HTTPS access. HTTP services continue to run for other phones that do not support HTTPS.

For information on provisioning Cisco Unified IP phones for secure access to web content using HTTPS, see HTTPS Provisioning for Cisco Unified IP Phones, on page 58.

For configuration examples, see Example for Configuring HTTPS Support for Cisco Unified CME, on page 79.

Configure Security

Configure the Cisco IOS Certification Authority

To configure a Cisco IOS Certification Authority (CA) on a local or external router, perform the following steps.

Note If you use a third-party CA, follow the provider's instructions instead of performing these steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http server**
4. **crypto pki server** *label*
5. **database level** {**minimal** | **names** | **complete**}
6. **database url** *root-url*
7. **lifetime certificate** *time*
8. **issuer-name CN=***label*

9. **exit**
10. **crypto pki trustpoint** *label*
11. **enrollment url** *ca-url*
12. **exit**
13. **crypto pki server** *label*
14. **grant auto**
15. **no shutdown**
16. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip http server** **Example:** `Router(config)# ip http server`	Enables the Cisco web-browser user interface on the local Cisco Unified CME router.
Step 4	**crypto pki server** *label* **Example:** `Router(config)# crypto pki server sanjose1`	Defines a label for the Cisco IOS CA and enters certificate-server configuration mode.
Step 5	**database level** {**minimal** | **names** | **complete**} **Example:** `Router(config-cs-server)# database level complete`	(Optional) Controls the type of data stored in the certificate enrollment database. • **minimal**—Enough information is stored only to continue issuing new certificates without conflict. This is the default value. • **names**—In addition to the minimal information given, the serial number and subject name of each certificate are also provided. • **complete**—In addition to the information given in the minimal and names levels, each issued certificate is written to the database. If you use this keyword, you must also specify an external TFTP server in which to store the data by using the **database url** command.

	Command or Action	Purpose
Step 6	**database url** *root-url* **Example:** `Router(config-cs-server)# database url nvram:`	(Optional) Specifies the location, other than NVRAM, where all database entries for the certificate server are to be written out. • Required if you configured the **complete** keyword with the **database level** command in the previous step. • *root-url*—URL that is supported by the Cisco IOS file system and where database entries are to be written out. If the CA is going to issue a large number of certificates, select an appropriate storage location like flash or other storage device to store the certificates. • When the storage location chosen is flash and the file system type on this device is Class B (LEFS), make sure to check free space on the device periodically and use the **squeeze** command to free the space used up by deleted files. This process may take several minutes and should be done during scheduled maintenance periods or off-peak hours.
Step 7	**lifetime certificate** *time* **Example:** `Router(config-cs-server) lifetime certificate 888`	(Optional) Specifies the lifetime, in days, of certificates issued by this Cisco IOS CA. • *time*—Number of days until a certificate expires. Range is 1 to 1825 days. Default is 365. The maximum certificate lifetime is 1 month less than the lifetime of the CA certificate. • Configure this command before the Cisco IOS CA is enabled by using the **no shutdown** command.
Step 8	**issuer-name CN=***label* **Example:** `Router(config-cs-server)# issuer-name CN=sanjose1`	(Optional) Specifies a distinguished name (DN) as issuer name for the Cisco IOS CA. • Default is already-configured label for the Cisco IOS CA. See Step 4, on page 20.
Step 9	**exit** **Example:** `Router(config-cs-server)# exit`	Exits certificate-server configuration mode.
Step 10	**crypto pki trustpoint** *label* **Example:** `Router(config)# crypto pki trustpoint sanjose1`	(Optional) Declares a trustpoint and enters ca-trustpoint configuration mode. • For local CA only. This command is not required for Cisco IOS CA on an external router. • If you must use a specific RSA key for the Cisco IOS CA, use this command to create your own trustpoint

	Command or Action	Purpose
		by using the same label to be used with the **crypto pki server** command. If the router sees a configured trustpoint with the same label as the crypto pki server, it uses this trustpoint and does not automatically create a trustpoint.
Step 11	**enrollment url** *ca-url* **Example:** `Router(config-ca-trustpoint)# enrollment url http://ca-server.company.com`	Specifies the enrollment URL of the issuing Cisco IOS CA. • For local Cisco IOS CA only. This command is not required for Cisco IOS CA on an external router. • *ca-url*—URL of the router on which the Cisco IOS CA is installed.
Step 12	**exit** **Example:** `Router(config-ca-trustpoint)# exit`	Exits ca-trustpoint configuration mode.
Step 13	**crypto pki server** *label* **Example:** `Router(config)# crypto pki server sanjose1`	Enters certificate-server configuration mode. • *label*—Name of the Cisco IOS CA being configured.
Step 14	**grant auto** **Example:** `Router(config-cs-server)# grant auto`	(Optional) Allows certificates to be issued automatically to any requester. • Default and recommended method is manual enrollment. • Use this command only when testing and building simple networks. Use the **no grant auto** command after configuration is complete to prevent certificates from being automatically granted.
Step 15	**no shutdown** **Example:** `Router(config-cs-server)# no shutdown`	(Optional) Enables the Cisco IOS CA. • Use this command only after you are finished configuring the Cisco IOS CA.
Step 16	**end** **Example:** `Router(config-cs-server)# end`	Returns to privileged EXEC mode.

Example

The following partial output from the **show running-config** command shows the configuration for a Cisco IOS CA named sanjose1 running on the local Cisco Unified CME router:

```
ip http server

crypto pki server sanjose1
 database level complete
 database url nvram:

crypto pki trustpoint sanjose1
 enrollment url http://ca-server.company.com

crypto pki server authority1
 no grant auto
 no shutdown
```

Obtain Certificates for Server Functions

The CA issues certificates for the following server functions:

- Cisco Unified CME—Requires a certificate for TLS sessions with phones.
- TFTP—Requires a key pair and certificate for signing configuration files.
- HTTPS—Requires a key pair and certificate for signing configuration files.
- CAPF—Requires a certificate for TLS sessions with phones.
- SAST—Required for signing the CTL file. We recommend creating two SAST certificates, one for primary use and one for backup.

To obtain a certificate for a server function, perform the following steps for each server function.



Note You can configure a different trustpoint for each server function or you can configure the same trustpoint for more than one server function as shown in Configuration Examples for Security, on page 64 at the end of this module.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint** *trustpoint-label*
4. **enrollment url** *url*
5. **revocation-check** *method1* [*method2* [*method3*]]
6. **rsakeypair** *key-label* [*key-size* [*encryption-key-size*]]
7. **exit**
8. **crypto pki authenticate** *trustpoint-label*
9. **crypto pki enroll** *trustpoint-label*
10. **exit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**crypto pki trustpoint** *trustpoint-label* **Example:** `Router(config)# crypto pki trustpoint capf`	Declares the trustpoint that the CA should use and enters ca-trustpoint configuration mode. • *trustpoint-label*—Label for server function being configured.
Step 4	**enrollment url** *url* **Example:** `Router(config-ca-trustpoint)# enrollment url http://ca-server.company.com`	Specifies the enrollment URL of the issuing CA. • *url*—URL of the router on which the issuing CA is installed.
Step 5	**revocation-check** *method1* [*method2* [*method3*]] **Example:** `Router(config-ca-trustpoint)# revocation-check none`	(Optional) Specifies the method to be used to check the revocation status of a certificate. • *method*—If a second and third method are specified, each subsequent method is used only if the previous method returns an error, such as a server being down. • **crl**—Certificate checking is performed by a certificate revocation list (CRL). This is the default behavior. • **none**—Certificate checking is not required. • **ocsp**—Certificate checking is performed by an Online Certificate Status Protocol (OCSP) server.
Step 6	**rsakeypair** *key-label* [*key-size* [*encryption-key-size*]] **Example:** `Router(config-ca-trustpoint)# rsakeypair capf 1024 1024`	(Optional) Specifies a key pair to use with a certificate. • *key-label*—Name of the key pair, which is generated during enrollment if it does not already exist or if the **auto-enroll regenerate** command is configured. • *key-size*—Size of the desired RSA key. If not specified, the existing key size is used. • *encryption-key-size*—Size of the second key, which is used to request separate encryption, signature keys, and certificates.

	Command or Action	Purpose
		• Multiple trustpoints can share the same key.
Step 7	**exit** **Example:** `Router(config-ca-trustpoint)# exit`	Exits ca-trustpoint configuration mode.
Step 8	**crypto pki authenticate** *trustpoint-label* **Example:** `Router(config)# crypto pki authenticate capf`	Retrieves the CA certificate, authenticates it, and checks the certificate fingerprint if prompted. • This command is optional if the CA certificate is already loaded into the configuration • *trustpoint-label*—Already-configured label for server function being configured.
Step 9	**crypto pki enroll** *trustpoint-label* **Example:** **`crypto pki enroll`** `trustpoint-label` `Router(config)# crypto pki enroll capf`	Enrolls with the CA and obtains the certificate for this trustpoint. • *trustpoint-label*—Already-configured label for server function being configured.
Step 10	**exit** **Example:** `Router(config)# exit`	Returns to privileged EXEC mode.

Example

The following partial output from the **show running-config** command show how to obtain certificates for a variety of server functions:

Obtaining a certificate for the CAPF server function

```
!configuring a trust point
 crypto pki trustpoint capf-server
 enrollment url http://192.168.1.1:80
 revocation-check none
!authenticate w/ the CA and download its certificate
crypto pki authenticate capf-server
! enroll with the CA and obtain this trustpoint's certificate
 crypto pki enroll capf-server
```

Obtaining a certificate for the Cisco Unified CME server function

```
crypto pki trustpoint cme-server
 enrollment url http://192.168.1.1:80
 revocation-check none

crypto pki authenticate cme-server
crypto pki enroll cme-server
```

Obtaining a certificate for the TFTP server function

```
crypto pki trustpoint tftp-server
 enrollment url http://192.168.1.1:80
 revocation-check none

crypto pki authenticate tftp-server
crypto pki enroll tftp-server
```

Obtaining a certificate for the first SAST server function (sast1)

```
crypto pki trustpoint sast1
  enrollment url http://192.168.1.1:80
 revocation-check none

 crypto pki authenticate sast1
crypto pki enroll sast1
```

Obtaining a certificate for the second SAST server function (sast2)

```
crypto pki trustpoint sast2
 enrollment url http://192.168.1.1:80
 revocation-check none

crypto pki authenticate sast2
crypto pki enroll sast2
```

Configure Telephony-Service Security Parameters

To configure security parameters for telephony service, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **telephony-service**
4. **secure-signaling trustpoint** *label*
5. **tftp-server-credentials trustpoint** *label*
6. **device-security-mode** {**authenticated** | **none** | **encrypted**}
7. **cnf-file perphone**
8. **load-cfg-file** *file-url* **alias** *file-alias* [**sign**] [**create**]
9. **server-security-mode** {**erase** | **non-secure** | **secure**}
10. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**telephony-service** **Example:** `Router(config)# telephony-service`	Enters telephony-service configuration mode.
Step 4	**secure-signaling trustpoint** *label* **Example:** `Router(config-telephony)# secure-signaling trustpoint cme-sccp`	Configures trustpoint to be used for secure signalling. • *label*—Name of a configured PKI trustpoint with a valid certificate to be used for TLS handshakes with IP phones on TCP port 2443.
Step 5	**tftp-server-credentials trustpoint** *label* **Example:** `Router(config-telephony)# tftp-server-credentials trustpoint cme-tftp`	Configures the TFTP server credentials (trustpoint) to be used for signing the configuration files. • *label*—Name of a configured PKI trustpoint with a valid certificate to be used to sign the phone configuration files. This can be the CAPF trustpoint that was used in the previous step or any trustpoint with a valid certificate
Step 6	**device-security-mode** {**authenticated** | **none** | **encrypted**} **Example:** `Router(config-telephony)# device-security-mode authenticated`	Enables security mode for endpoints. • **authenticated**—Instructs device to establish a TLS connection with no encryption. There is no Secure Real-Time Transport Protocol (SRTP) in the media path. • **none**—SCCP signaling is not secure. This is the default. • **encrypted**—Instructs device to establish an encrypted TLS connection to secure media path using SRTP. • This command can also be configured in ephone configuration mode. The value set in ephone configuration mode has priority over the value set in telephony-service configuration mode.

	Command or Action	Purpose
Step 7	**cnf-file perphone** **Example:** `Router(config-telephony)# cnf-file perphone`	Specifies that the system generate a separate XML configuration file for each IP phone. • Separate configuration files for each endpoint are required for security.
Step 8	**load-cfg-file** *file-url* **alias** *file-alias* [**sign**] [**create**] **Example:** `Router(config-telephony)# load-cfg-file slot0:Ringlist.xml alias Ringlist.xml sign create`	(Optional) Signs configuration files that are not created by Cisco Unified CME. Also loads the signed and unsigned versions of a file on the TFTP server. • *file-url*—Complete path of a configuration file in a local directory. • **alias** *file-alias*—Alias name of the file to be served on the TFTP server. • **sign**—(Optional) The file needs to be digitally signed and served on the TFTP server. • **create**—(Optional) Creates the signed file in the local directory. • The first time that you use this command for each file, use the **create** and **sign** keywords. The **create** keyword is not maintained in the running configuration to prevent signed files from being recreated during every reload. • To serve an already-signed file on the TFTP server, use this command without the **create** and **sign** keywords.
Step 9	**server-security-mode** {**erase** | **non-secure** | **secure**} **Example:** `Router(config-telephony)# server-security-mode non-secure`	(Optional) Changes the security mode of the server. • **erase**—Deletes the CTL file. • **non-secure**—Nonsecure mode. • **secure**—Secure mode. • This command has no impact until the CTL file is initially generated by the CTL client. When the CTL file is generated, the CTL client automatically sets server security mode to secure.
Step 10	**end** **Example:** `Router(config-ephone)# end`	Returns to privileged EXEC mode.

Verify Telephony-Service Security Parameters

Procedure

Step 1 **show telephony-service security-info**

Use this command to display the security-related information that is configured in telephony-service configuration mode.

Example:

```
Router# show telephony-service security-info

Skinny Server Trustpoint for TLS: cme-sccp
TFTP Credentials Trustpoint: cme-tftp
Server Security Mode: Secure
Global Device Security Mode: Authenticated
```

Step 2 **show running-config**

Use this command to display the running configuration to verify telephony and per-phone security configuration.

Example:

```
Router# show running-config

telephony-service
  secure-signaling trustpoint cme-sccp
  server-security-mode secure
  device-security-mode authenticated
  tftp-server-credentials trustpoint cme-tftp
  .
  .
  .
```

Configure the CTL Client

Perform one of the following tasks, depending upon your network configuration:

- Configure the CTL Client on a Cisco Unified CME Router, on page 29
- Configure the CTL Client on a Router That is Not a Cisco Unified CME Router, on page 32

Configure the CTL Client on a Cisco Unified CME Router

To configure a CTL client for creating a list of known, trusted certificates and tokens on a local Cisco Unified CME router, perform the following steps.

Note If you have primary and secondary Cisco Unified CME routers, you can configure the CTL client on either one of them.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ctl-client**
4. **sast1 trustpoint** *label*
5. **sast2 trustpoint** *label*
6. **server** {**capf** | **cme** | **cme-tftp** | **tftp**} *ip-address* **trustpoint** *trustpoint-label*
7. **server cme** *ip-address* **username** *name-string* **password** {**0** | **1**} *password-string*
8. **regenerate**
9. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ctl-client** **Example:** `Router(config)# ctl-client`	Enters CTL-client configuration mode.
Step 4	**sast1 trustpoint** *label* **Example:** `Router(config-ctl-client)# sast1 trustpoint sast1tp`	Configures credentials for the primary SAST. • *label*- Name of SAST1 trustpoint. **Note** SAST1 and SAST2 certificates must be different from each other. The CTL file is always signed by SAST1. The SAST2 credentials are included in the CTL file so that if the SAST1 certificate is compromised, the file can be signed by SAST2 to prevent phones from being reset to the factory default.
Step 5	**sast2 trustpoint** *label* **Example:** `Router(config-ctl-client)# sast2 trustpoint`	Configures credentials for the secondary SAST. • *label* - name of SAST2 trustpoint. **Note** SAST1 and SAST2 certificates must be different from each other. The CTL file is always signed by SAST1. The SAST2 credentials are included in the CTL file so that if

	Command or Action	Purpose
		the SAST1 certificate is compromised, the file can be signed by SAST2 to prevent phones from being reset to the factory default.
Step 6	**server** {**capf** | **cme** | **cme-tftp** | **tftp**} *ip-address* **trustpoint** *trustpoint-label* **Example:** `Router(config-ctl-client)# server capf 10.2.2.2 trustpoint capftp`	Configures a trustpoint for each server function that is running locally on the Cisco Unified CME router. • *ip-address* - IP address of the Cisco Unified CME router. If there are multiple network interfaces, use the interface address in the local LAN to which the phones are connected. • **trustpoint** *trustpoint-label*- Name of the PKI trustpoint for the server function being configured. • Repeat this command for server each function that is running locally on the Cisco Unified CME router.

Step 7	**server cme** *ip-address* **username** *name-string* **password** {**0** | **1**} *password-string*
	Example:
	`Router(config-ctl-client)# server cme 10.2.2.2 username user3 password 0 38h2KL`
		(Optional) Provides information for another Cisco Unified CME router (primary or secondary) in the network.
		• *ip-address*- IP address of the othe Cisco Unified CME router.
		• **username** *name-string*- Username that is configured on the CTL provider.
		• **password**- Defines the way that you want the password to appear in **show** command output and not to the way that you enter the password.
		  • **0**- Not encrypted.
		  • **1**- Encrypted using Message Digest 5 (MD5).
		• *password-string*- Administrative password of the CTL provider running on the remote Cisco Unified CME router.

Step 8	**regenerate** **Example:** `Router(config-ctl-client)# regenerate`	Creates a new CTLFile.tlv after you make changes to the CTL client configuration.
Step 9	**end** **Example:** `Router(config-ctl-client)# end`	Returns to privileged EXEC mode.

Examples

The following sample output from the **show ctl-client** command displays the trustpoints in the system:

```
Router# show ctl-client

CTL Client Information
---------------------------
  SAST 1 Certificate Trustpoint: cmeserver
         SAST 1 Certificate Trustpoint: sast2
         List of Trusted Servers in the CTL
                CME     10.1.1.1        cmeserver
                TFTP    10.1.1.1        cmeserver
                CAPF    10.1.1.1        cmeserver
```

What to do next

You are finished configuring the CTL client. See Configure the CAPF Server, on page 34.

Configure the CTL Client on a Router That is Not a Cisco Unified CME Router

To configure a CTL client on a stand-alone router that is not a Cisco Unified CME router, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ctl-client**
4. **sast1 trustpoint** *label*
5. **sast2 trustpoint** *label*
6. **server cme** *ip-address* **username** *name-string* **password** {**0** | **1**} *password-string*
7. **regenerate**
8. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ctl-client** **Example:** `Router(config)# ctl-client`	Enters ctl-client configuration mode.
Step 4	**sast1 trustpoint** *label*	Configures credentials for the primary SAST.

	Command or Action	Purpose
	Example: `Router(config-ctl-client)# sast1 trustpoint sast1tp`	• *label*—Name of SAST1 trustpoint. **Note** SAST1 and SAST2 certificates must be different from each other but either of them may use the same certificate as the Cisco Unified CME router to conserve memory. The CTL file is always signed by SAST1. The SAST2 credentials are included in the CTL file so that if the SAST1 certificate is compromised, the file can be signed by SAST2 to prevent phones from being reset to the factory default.
Step 5	**sast2 trustpoint** *label* **Example:** `Router(config-ctl-client)# sast2 trustpoint`	Configures credentials for the secondary SAST. • *label*—name of SAST2 trustpoint. **Note** SAST1 and SAST2 certificates must be different from each other but either of them may use the same certificate as the Cisco Unified CME router to conserve memory. The CTL file is always signed by SAST1. The SAST2 credentials are included in the CTL file so that if the SAST1 certificate is compromised, the file can be signed by SAST2 to prevent phones from being reset to the factory default.

Step 6	**server cme** *ip-address* **username** *name-string* **password** {**0** | **1**} *password-string*
	Example:
	`Router(config-ctl-client)# server cme 10.2.2.2 username user3 password 0 38h2KL`
		(Optional) Provides information about another Cisco Unified CME router (primary or secondary) in the network, if one exists.
		• *ip-address*—IP address of the other Cisco Unified CME router.
		• **username** *name-string*—Username that is configured on the CTL provider.
		• **password**—Encryption status of the password string.
		  • **0**—Not encrypted.
		  • **1**—Encrypted using Message Digest 5 (MD5).
		Note This option refers to the way that you want the password to appear in show command output and not to the way that you enter the password in this command.
		• *password-string*—Administrative password of the CTL provider running on the remote Cisco Unified CME router.

Step 7	**regenerate** **Example:**	Creates a new CTLFile.tlv after you make changes to the CTL client configuration.

	Command or Action	Purpose
	`Router(config-ctl-client)# regenerate`	
Step 8	**end** **Example:** `Router(config-ctl-client)# end`	Returns to privileged EXEC mode.

Examples

The following sample output from the **show ctl-client** command displays the trustpoints in the system:

```
Router# show ctl-client

CTL Client Information
---------------------------
 SAST 1 Certificate Trustpoint: cmeserver
        SAST 1 Certificate Trustpoint: sast2
        List of Trusted Servers in the CTL
                CME     10.1.1.1        cmeserver
                TFTP    10.1.1.1        cmeserver
                CAPF    10.1.1.1        cmeserver
```

Configure the CAPF Server

A certificate must be obtained for the CAPF server so that it can establish a TLS session with the phone during certificate operation. The CAPF server can install, fetch, or delete locally significant certificates (LSCs) on security-enabled phones. To enable the CAPF server on the Cisco Unified CME router, perform the following steps.

Tip When you use the CAPF server to install phone certificates, arrange to do so during a scheduled period of maintenance. Generating many certificates at the same time may cause call-processing interruptions.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **capf-server**
4. **trustpoint-label** *label*
5. **cert-enroll-trustpoint** *label* **password** {**0** | **1**} *password-string*
6. **source-addr** *ip-address*
7. **auth-mode** {**auth-string** | **LSC** | **MIC** | **none** | **null-string**}
8. **auth-string** {**delete** | **generate**} {**all** | *ephone-tag*} [*digit-string*]
9. **phone-key-size** {**512** | **1024** | **2048**}
10. **port** *tcp-port*
11. **keygen-retry** *number*
12. **keygen-timeout** *minutes*
13. **cert-oper** {**delete all** | **fetch all** | **upgrade all**}

14. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**capf-server** **Example:** `Router(config)# capf-server`	Enters capf-server configuration mode.
Step 4	**trustpoint-label** *label* **Example:** `Router(config-capf-server)# trustpoint-label tp1`	Specifies the label for the trustpoint. • *label*—Name of trustpoint whose certificate is to be used for TLS connection between the CAPF server and the phone.
Step 5	**cert-enroll-trustpoint** *label* **password** {**0** | **1**} *password-string* **Example:** `Router(config-capf-server)# cert-enroll-trustpoint ra1 password 0 x8oWiet`	Enrolls the CAPF with the CA (or RA, if the CA is not local to the Cisco Unified CME router). • *label*—PKI trustpoint label for CA and RA that was previously configured by using the **crypto pki trustpoint** command in global configuration mode. • **password**—Encryption status of the password string. • *password-string*—Password to use for certificate enrollment. This password is the revocation password that is sent along with the certificate request to the CA.
Step 6	**source-addr** *ip-address* **Example:** `Router(config-capf-server)# source addr 10.10.10.1`	Defines the IP address of the CAPF server on the Cisco Unified CME router.
Step 7	**auth-mode** {**auth-string** | **LSC** | **MIC** | **none** | **null-string**} **Example:** `Router(config-capf-server)# auth-mode auth-string`	Specifies the type of authentication mode for CAPF sessions to verify endpoints that request certificates. • **auth-string**—The phone user enters a special authentication string at the phone. The string is provided to the user by the system administrator and

	Command or Action	Purpose
		is configured using the **auth-string generate** command. • **LSC**—The phone provides its LSC for authentication, if one exists. • **MIC**—The phone provides its MIC for authentication, if one exists. If this option is chosen, the MIC s issuer certificate must be imported into a PKI trustpoint. • **none**—No certificate upgrade is initiated. This is the default. • **null-string**—No authentication.
Step 8	**auth-string** {**delete** | **generate**} {**all** | *ephone-tag*} [*digit-string*] **Example:** `Router(config-capf-server)# auth-string generate all`	(Optional) Creates or removes authentication strings for one or all secure phones. • Use this command if the **auth-string** keyword is specified in the previous step. Strings become part of the ephone configuration. • **delete**—Remove authentication strings for the specified secure devices. • **generate**—Create authentication strings for the specified secure devices. • **all**—All phones. • *ephone-tag*—identifier for the ephone to receive the authentication string. • *digit-string*—Digits that phone user must dial for CAPF authentication. Length of string is 4 to 10 digits that can be pressed on the keypad. If this value is not specified, a random string is generated for each phone. • You can also define an authentication string for an individual SCCP IP phone by using the **capf-auth-str** command in ephone configuration mode.
Step 9	**phone-key-size** {**512** | **1024** | **2048**} **Example:** `Router(config-capf-server)# phone-key-size 2048`	(Optional) Specifies the size of the RSA key pair that is generated on the phone for the phone s certificate, in bits. • **512**—512. • **1024**—1024. This is the default. • **2048**—2048.
Step 10	**port** *tcp-port* **Example:** `Router(config-capf-server)# port 3804`	(Optional) Defines the TCP port number on which the CAPF server listens for socket connections from the phones.

	Command or Action	Purpose
		• *tcp-port*—TCP port number. Range is 2000 to 9999. Default is 3804.
Step 11	**keygen-retry** *number* **Example:** `Router(config-capf-server)# keygen-retry 5`	(Optional) Specifies the number of times that the server sends a key generation request. • *number*—Number of retries. Range is 0 to 100. Default is 3.
Step 12	**keygen-timeout** *minutes* **Example:** `Router(config-capf-server)# keygen-timeout 45`	(Optional) Specifies the amount of time that the server waits for a key generation response from the phone. • *minutes*—Number of minutes before the generation process times out. Range is 1 to 120. Default is 30.
Step 13	**cert-oper** {**delete all** | **fetch all** | **upgrade all**} **Example:** `Router(config-capf-server)# cert-oper upgrade all`	(Optional) Initiates the indicated certificate operation on all configured endpoints in the system. • **delete all**—Remove all phone certificates. • **fetch all**—Retrieve all phone certificates for troubleshooting. • **upgrade all**—Upgrade all phone certificates. • This command can also be configured in ephone configuration mode to initiate certificate operations on individual phones. This command in ephone configuration mode has priority over this command in CAPF-server configuration mode.
Step 14	**end** **Example:** `Router(config-capf-server)# end`	Returns to privileged EXEC mode.

Verify the CAPF Server

Use the **show capf-server summary** command to display CAPF-server configuration information.

```
Router# show capf-server summary

CAPF Server Configuration Details
        Trustpoint for TLS With Phone: tp1
        Trustpoint for CA operation: ra1
        Source Address: 10.10.10.1
        Listening Port: 3804
        Phone Key Size: 1024
        Phone KeyGen Retries: 3
        Phone KeyGen Timeout: 30 minutes
```

Configure Ephone Security Parameters

To configure security parameters for individual phones, perform the following steps for each phone.

Before you begin

- Phones to be configured for security must be configured for basic calling in Cisco Unified CME. For configuration information, see Configure Phones to Make Basic Call.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ephone** *phone-tag*
4. **capf-ip-in-cnf**
5. **device-security-mode** {**authenticated** | **none** | **encrypted** }
6. **codec** {**g711ulaw** | **g722r64** | **g729r8** [**dspfarm-assist**] }
7. **capf-auth-str** *digit-string*
8. **cert-oper** {**delete** | **fetch** | **upgrade**} **auth-mode** {**auth-string** | **LSC** | **MIC** | **null-string**}
9. **reset**
10. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ephone** *phone-tag* **Example:** `Router(config)# ephone 24`	Enters ephone configuration mode. • *phone-tag*—Unique identifier of phone to be configured.
Step 4	**capf-ip-in-cnf** **Example:** `Router(config-ephone)# capf-ip-in-cnf`	(Optional) Enables the CAPF Server IP Address to be added to the CNF file for an SCCP phone. Upon successful registration, the SCCP phone downloads the LSC from the CAPF server. This CLI command is optional and required only if the phone has to register, download, and authenticate with the LSC.

	Command or Action	Purpose
Step 5	**device-security-mode** {**authenticated** | **none** | **encrypted** } **Example:** `Router(config-ephone)# device-security-mode authenticated`	(Optional) Enables security mode for an individual SCCP IP phone. • **authenticated**—Instructs device to establish a TLS connection with no encryption. There is no Secure Real-Time Transport Protocol (SRTP) in the media path. • **none**—SCCP signaling is not secure. This is the default. • **encrypted**—Instructs device to establish an encrypted TLS connection to secure media path using SRTP. • This command can also be configured in telephony-service configuration mode. The value set in ephone configuration mode has priority over the value set in telephony-service configuration mode.
Step 6	**codec** {**g711ulaw** | **g722r64** | **g729r8** [**dspfarm-assist**] } **Example:** `Router(config-ephone)# codec g711ulaw dspfarm-assist`	(Optional) Sets the security mode for SCCP signaling for a phone communicating with the Cisco Unified CME router. • **dspfarm-assist**—Required for secure transcoding with Cisco Unified CME. Causes the system to attempt to use DSP Farm resources for transcoding the segment between the phone and the Cisco Unified CME router if G.711 is negotiated for the call. This keyword is ignored if the SCCP endpoint type is ATA, VG224, or VG248.
Step 7	**capf-auth-str** *digit-string* **Example:** `Router(config-ephone)# capf-auth-str 2734`	(Optional) Defines a string to use as a personal identification number (PIN) for CAPF authentication. **Note** For instructions on how to enter the string on a phone, see Enter the Authentication String on the Phone, on page 47. • *digit-string*—Digits that the phone user must dial for CAPF authentication. The length of string is 4 to 10 digits. • This command can also be configured in telephony-service configuration mode. The value set in ephone configuration mode has priority over the value set in telephony-service configuration mode. • You can also define a PIN for CAPF authentication by using the**auth-string** command in CAPF-server configuration mode.

	Command or Action	Purpose
Step 8	**cert-oper** {**delete** | **fetch** | **upgrade**} **auth-mode** {**auth-string** | **LSC** | **MIC** | **null-string**} **Example:** `Router(config-ephone)# cert-oper upgrade auth-mode auth-string`	(Optional) Initiates the indicated certificate operation on the ephone being configured. • **delete**—Removes the phone certificate. • **fetch**—Retrieves the phone certificate for troubleshooting. • **upgrade**—Upgrades the phone certificate. • **auth-mode**—Type of authentication to use during CAPF sessions to verify endpoints that request certificates. • **auth-string**—Authentication string to be entered on the phone by the phone user. Use the **capf-auth-str** command to configure the auth-string. For configuration information, see Enter the Authentication String on the Phone, on page 47. • **LSC**—Phone provides its phone certificate for authentication. Precedence is given to an LSC if one exists. • **MIC**—Phone provides its phone certificate for authentication. Precedence is given to an MIC if one exists. MIC s issuer certificate must be imported into a PKI trustpoint. For information, see Manually Import the MIC Root Certificate, on page 48. • **null-string**—No authentication. • This command can also be configured in CAPF-server configuration mode to initiate certificate operations at a global level. This command in ephone configuration mode has priority over this command in CAPF-server configuration mode. • You can also use the **auth-mode** command in CAPF-server configuration mode to configure authentication at a global level.
Step 9	**reset** **Example:** `Router(config-ephone)# reset`	Performs a complete reboot of the phone.
Step 10	**end** **Example:** `Router(config-ephone)# end`	Returns to privileged EXEC mode.

Verify Ephone Security Parameters

Procedure

Use the **show capf-server auth-string** command to display configured authentication strings (PINs) that users enter at the phone to establish CAPF authentication.

Example:

```
Router# show capf-server auth-string

  Authentication Strings for configured Ephones
  Mac-Addr        Auth-String
  --------        -----------
  000CCE3A817C    2734
  001121116BDD    922
  000D299D50DF    9182
  000ED7B10DAC    3114
  000F90485077    3328>
  0013C352E7F1    0678
```

What to do next

- When you have more than one Cisco Unified CME router in your network, you must configure a CTL provider on each Cisco Unified CME router that is not running the CTL client. To configure a CTL provider on each Cisco Unified CME router on which the CTL client is not running, see Configure the CTL Provider, on page 41.
- If the CA is a third-party CA or if the Cisco IOS CA is on a Cisco IOS router external to the Cisco Unified CME router, you must configure an RA to issue certificates to phones. For information, see Configure the Registration Authority, on page 44.
- If the specified authentication mode for the CAPF session is authentication-string, you must enter an authentication string on each phone that is receiving an updated LSC. For information, see Enter the Authentication String on the Phone, on page 47.
- If the specified authentication mode for the CAPF session is MIC, the MIC's issuer certificate must be imported into a PKI trustpoint. For information, see Manually Import the MIC Root Certificate, on page 48.
- To configure Media Encryption, see Configure Media Encryption (SRTP) in Cisco Unified CME, on page 51.

Configure the CTL Provider

When you have more than one Cisco Unified CME router in your network, you must configure a CTL provider on each Cisco Unified CME router that is not running the CTL client. To configure a CTL provider on each Cisco Unified CME router on which the CTL client is not running, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **credentials**
4. **ip source-address** [*ip-address* [**port** [*port-number*]]]
5. **trustpoint** *trustpoint-label*
6. **ctl-service admin username secret {0 | 1 }** *password*- string
7. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**credentials** **Example:** `Router(config)# credentials`	Enters credentials-interface mode to configure a CTL provider.
Step 4	**ip source-address** [*ip-address* [**port** [*port-number*]]] **Example:** `Router(config-credentials)# ip source-address 172.19.245.1 port 2444`	identifies the local router on which this CTL provider is being configured. • *ip-address*—Typically one of the addresses of the Ethernet port of the router. • **port** *port-number*—TCP port for credentials service communication. Default is 2444 and we recommend that you use the default value.
Step 5	**trustpoint** *trustpoint-label* **Example:** `Router(config-credentials)# trustpoint ctlpv`	Configures the trustpoint. • *trustpoint-label*—Name of CTL provider trustpoint to be used for TLS sessions with the CTL client.
Step 6	**ctl-service admin username secret {0 | 1 }** *password*- string **Example:** `Router(config-credentials)# ctl-service admin user4 secret 0 c89L8o`	Specifies a username and password to authenticate the CTL client when it connects to retrieve the credentials during the CTL protocol. • *username*—Name that will be used to authenticate the client.

	Command or Action	Purpose

		• **secret**—Character string for login authentication and whether the string should be encrypted when it is stored in the running configuration.
		  • **0**—Not encrypted.
		  • **1**—Encrypted using Message Digest 5 (MD5).
		• *password-string*—Character string for login authentication.

Step 7	**end** **Example:** `Router(config-credentials)# end`	Returns to privileged EXEC mode.

Verify the CTL Provider

Procedure

Use the **show credentials** command to display credentials settings.

Example:

```
Router# show credentials

Credentials IP: 172.19.245.1
Credentials PORT: 2444
Trustpoint: ctlpv
```

What to do next

- If the CA is a third-party CA or if the Cisco IOS CA is on a Cisco IOS router external to the Cisco Unified CME router, you must configure an RA to issue certificates to phones. For information, see Configure the Registration Authority, on page 44.
- If the specified authentication mode for the CAPF session is authentication-string, you must enter an authentication string on each phone that is receiving an updated LSC. For information, see Enter the Authentication String on the Phone, on page 47.
- If the specified authentication mode for the CAPF session is MIC, the MIC's issuer certificate must be imported into a PKI trustpoint. For information, see Manually Import the MIC Root Certificate, on page 48.
- To configure Media Encryption, see Configure Media Encryption (SRTP) in Cisco Unified CME, on page 51.

Configure the Registration Authority

A registration authority (RA) is the authority charged with recording or verifying some or all of the data required for the CA to issue certificates. In many cases the CA undertakes all of the RA functions itself, but where a CA operates over a wide geographical area or when there is security concern over exposing the CA at the edge of the network, it may be advisable to delegate some of the tasks to an RA and let the CA concentrate on its primary tasks of signing certificates.

You can configure a CA to run in RA mode. When the RA receives a manual or Simple Certificate Enrollment Protocol (SCEP) enrollment request, the administrator can either reject or grant it on the basis of local policy. If the request is granted, it is forwarded to the issuing CA, and the CA automatically generates the certificate and returns it to the RA. The client can later retrieve the granted certificate from the RA.

To configure an RA, perform the following steps on the Cisco Unified CME router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint** *label*
4. **enrollment url** *ca-url*
5. **revocation-check** *method1* [*method2* [*method3*]]
6. **serial-number** [**none**]
7. **rsakeypair** *key-label* [*key-size* [*encryption-key-size*]]
8. **exit**
9. **crypto pki server** *label*
10. **mode ra**
11. **lifetime certificate** *time*
12. **grant auto**
13. **no shutdown**
14. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**crypto pki trustpoint** *label* **Example:**	Declares the trustpoint that your RA mode certificate server should use and enters CA-trustpoint configuration mode. • *label*—Name for the trustpoint and RA.

	Command or Action	Purpose
	`Router(config)# crypto pki trustpoint ra12`	**Tip** This label is also required for the **cert-enroll-trustpoint** command when you set up the CA proxy. See Configure the CAPF Server, on page 34.
Step 4	**enrollment url** *ca-url* **Example:** `Router(config-ca-trustpoint)# enrollment url http://ca-server.company.com`	Specifies the enrollment URL of the issuing CA (root CA). • *ca-url*—URL of the router on which the root CA has been installed.
Step 5	**revocation-check** *method1* [*method2* [*method3*]] **Example:** `Router(config-ca-trustpoint)# revocation-check none`	(Optional) Checks the revocation status of a certificate and specifies one or more methods to check the status. If a second and third method are specified, each method is used only if the previous method returns an error, such as a server being down. Valid values for *methodn* are as follows: • **crl**—Certificate checking is performed by a certificate revocation list (CRL). This is the default behavior. • **none**—Certificate checking is not required. • **ocsp**—Certificate checking is performed by an Online Certificate Status Protocol (OCSP) server.
Step 6	**serial-number** [**none**] **Example:** `Router(config-ca-trustpoint)# serial-number`	(Optional) Specifies whether the router serial number should be included in the certificate request. When this command is not used, you are prompted for the serial number during certificate enrollment. • **none**—(Optional) A serial number is not included in the certificate request.
Step 7	**rsakeypair** *key-label* [*key-size* [*encryption-key-size*]] **Example:** `Router(config-ca-trustpoint)# rsakeypair exampleCAkeys 1024 1024`	(Optional) Specifies an RSA key pair to use with a certificate. • *key-label*—Name of the key pair, which is generated during enrollment if it does not already exist or if the **auto-enroll regenerate** command is used. • *key-size*—(Optional) Size of the desired RSA key. If not specified, the existing key size is used. • *encryption-key-size*—(Optional) Size of the second key, which is used to request separate encryption, signature keys, and certificates. • Multiple trustpoints can share the same key.
Step 8	**exit** **Example:**	Exits ca-trustpoint configuration mode.

	Command or Action	Purpose
	`Router(config-ca-trustpoint)# exit`	
Step 9	**crypto pki server** *label* **Example:** `Router(config)# crypto pki server ra12`	Defines a label for the certificate server and enters certificate-server configuration mode. • *label*—Name for the trustpoint and RA. Use the same label that you previously created as a trustpoint and RA in Step 3, on page 44.
Step 10	**mode ra** **Example:** `Router(config-cs-server)# mode ra`	Places the PKI server into certificate-server mode for the RA.
Step 11	**lifetime certificate** *time* **Example:** `Router(config-cs-server)# lifetime certificate 1800`	(Optional) Specifies the lifetime, in days, of a certificate. • *time*—Number of days until the certificate expires. Range is 1 to 1825. Default is 365. The maximum certificate lifetime is 1 month less than the lifetime of the CA certificate. • This command must be used before the server is enabled with the **no shutdown** command.
Step 12	**grant auto** **Example:** `Router(config-cs-server)# grant auto`	Allows a certificate to be issued automatically to any requester. • Configure this command only during enrollment when testing and building simple networks. • As a security best practice, use the **no grant auto** command to disable this functionality after configuration so that certificates are not continually granted.
Step 13	**no shutdown** **Example:** `Router(config-cs-server)# no shutdown`	(Optional) Enables the certificate server. • When prompted, provide input regarding acceptance of the CA certificate, the router certificate, the challenge password, and a password for protecting the private key. • Use this command only after you have completely configured your certificate server.
Step 14	**end** **Example:** `Router(config-cs-server)# end`	Returns to privileged EXEC mode.

What to do next

- When you have more than one Cisco Unified CME router in your network, you must configure a CTL provider on each Cisco Unified CME router that is not running the CTL client. To configure a CTL provider on each Cisco Unified CME router on which the CTL client is not running, see Configure the CTL Provider, on page 41.
- If the specified authentication mode for the CAPF session is authentication-string, you must enter an authentication string on each phone that is receiving an updated LSC. For information, see Enter the Authentication String on the Phone, on page 47.
- If the specified authentication mode for the CAPF session is MIC, the MIC s issuer certificate must be imported into a PKI trustpoint. For information, see Manually Import the MIC Root Certificate, on page 48.
- To configure Media Encryption, see Configure Media Encryption (SRTP) in Cisco Unified CME, on page 51.

Enter the Authentication String on the Phone

This procedure is required only for the one-time installation of an LSC on a phone and only if you configured the authentication mode for the CAPF session as authentication-string. The authentication string must be communicated to the phone user so that it can be entered on the phone before the LSC is installed.

Note You can list authentication strings for phones by using the **show capf-server auth-string** command.

Restriction

- Authentication string applies for one-time use only.

Before you begin

- Signed image exists on the IP phone; see the Cisco Unified IP phone administration documentation that supports your phone model.
- IP phone is registered in Cisco Unified CME.
- CAPF certificate exists in the CTL file. For information, see Configure the CTL Client, on page 29.
- Authentication string to be entered is configured using **auth-string** command in CAPF-server configuration mode or the **capf-auth-str** command in ephone configuration mode. For information, see Configure Telephony-Service Security Parameters, on page 26.
- The **device-security-mode** command is configured using the **none** keyword. For information, see Configure Telephony-Service Security Parameters, on page 26.

Procedure

Step 1 Press the **Settings** button. On the Cisco Unified IP Phone 7921, press **Down Arrow** to access the **Settings** menu.

Step 2 If the configuration is locked, press ****#** (asterisk, asterisk, pound sign) to unlock it.

Step 3 Scroll down the **Settings** menu. Highlight Security Configuration and press the **Select** softkey.

Step 4 Scroll down the **Security Configuration** menu. Highlight LSC and press the **Update** softkey. On the Cisco Unified IP Phone 7921, press ****#** to unlock the Security Configuration menu.

Step 5 When prompted for the authentication string, enter the string provided by the system administrator and press the **Submit** softkey.

The phone installs, updates, deletes, or fetches the certificate, depending on the CAPF configuration.

You can monitor the progress of the certificate operation by viewing the messages that display on the phone. After you press **Submit**, the message "Pending" appears under the LSC option. The phone generates the public and private key pair and displays the information on the phone. When the phone successfully completes the process, the phone displays a successful message. If the phone displays a failure message, you entered the wrong authentication string or did not enable the phone for upgrade.

You can stop the process by choosing Stop at any time.

Step 6 Verify that the certificate was installed on the phone. From the **Settings** menu on the phone screen, choose **Model Information** and then press the **Select** softkey to display the Model Information.

Step 7 Press the navigation button to scroll to LSC. The value for this item indicates whether LSC is Installed or Not Installed.

What to do next

- When you have more than one Cisco Unified CME router in your network, you must configure a CTL provider on each Cisco Unified CME router that is not running the CTL client. To configure a CTL provider on each Cisco Unified CME router on which the CTL client is not running, see Configure the CTL Provider, on page 41.
- If the CA is a third-party CA or if the Cisco IOS CA is on a Cisco IOS router external to the Cisco Unified CME router, you must configure an RA to issue certificates to phones. For information, see Configure the Registration Authority, on page 44 .
- If the specified authentication mode for the CAPF session is MIC, the MIC's issuer certificate must be imported into a PKI trustpoint. For information, see Manually Import the MIC Root Certificate, on page 48.
- To configure Media Encryption, see Configure Media Encryption (SRTP) in Cisco Unified CME, on page 51.

Manually Import the MIC Root Certificate

The MIC root certificate must be present in the Cisco Unified CME router to allow Cisco Unified CME to authenticate the MIC that is presented to it. To manually import the MIC root certificate on the Cisco Unified CME router, perform the following steps for each type of phone that requires a MIC for authentication.

Before you begin

One of the following must be true before you perform this task:

- The **device-security-mode** command is configured using the **none** keyword. For information, see Configure Telephony-Service Security Parameters, on page 26.
- MIC is the specified authentication mode for phone authentication during a CAPF session.
- A phone's MIC, rather than an LSC, is used to establish the TLS session for SCCP signaling.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint** *name*
4. **revocation-check none**
5. **enrollment terminal**
6. **exit**
7. **crypto pki authenticate** *name*
8. Download the four MIC root certificate files. Cut and paste the appropriate text for each certificate. Accept the certificates.
9. **exit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**crypto pki trustpoint** *name* **Example:** `Router(config)# crypto pki trustpoint sanjose1`	Declares the CA that your router should use and enters ca-trustpoint configuration mode. • *name*—Already-configured label for the CA.
Step 4	**revocation-check none** **Example:** `Router(ca-trustpoint)# revocation-check none`	Specifies that revocation check is not performed and the certificate is always accepted.
Step 5	**enrollment terminal** **Example:**	Specifies manual (copy-and-paste) certificate enrollment.

	Command or Action	Purpose
	`Router(ca-trustpoint)# enrollment terminal`	
Step 6	**exit** **Example:** `Router(ca-trustpoint)# exit`	Exits ca-trustpoint configuration mode.
Step 7	**crypto pki authenticate** *name* **Example:** `Router(config)# crypto pki authenticate sanjose1`	Authenticates the CA by getting the certificate from the CA. • *name*- Already-configured label for the CA.
Step 8	Download the four MIC root certificate files. Cut and paste the appropriate text for each certificate. Accept the certificates.	**a.** Click on the link to the certificate: The certificates are available at the following links: • CAP-RTP-001: http://www.cisco.com/security/pki/certs/CAP-RTP-001.cer • CAP-RTP-002: http://www.cisco.com/security/pki/certs/CAP-RTP-002.cer • CMCA: http://www.cisco.com/security/pki/certs/cmca.cer • CiscoRootCA2048: http://www.cisco.com/security/pki/certs/crca2048.cer **b.** When the **Downloading Certificate** dialog window opens, select the option to view the certificate. Do not install the certificate. **c.** Select the **Detail** tab on top. **d.** Click **Export** on the bottom and save the certificate into a file. **e.** Open the file with WordPad. **f.** Cut and paste the text between -----BEGIN CERTIFICATE----- and -----END CERTIFICATE----- into the IOS console. **g.** When prompted, press **Enter** and type **`quit`**. After pasting the certificate, press **Enter** and type **`quit`** on a line by itself. **h.** Enter **`y`** to accept the certificate. The system responds to the pasted certificate text by providing the MD5 and SHA1 fingerprints, and asks whether you accept the certificate. Enter **`y`** to accept the certificate or **`n`** to reject it. **i.** Repeat steps a. through h. for each certificate.

	Command or Action	Purpose
Step 9	**exit** **Example:** `Router(config)# exit`	Returns to privileged EXEC mode.

What to do next

- When you have more than one Cisco Unified CME router in your network, you must configure a CTL provider on each Cisco Unified CME router that is not running the CTL client. To configure a CTL provider on each Cisco Unified CME router on which the CTL client is not running, see Configure the CTL Provider, on page 41.
- If the CA is a third-party CA or if the Cisco IOS CA is on a Cisco IOS router external to the Cisco Unified CME router, you must configure an RA to issue certificates to phones. For information, see Configure the Registration Authority, on page 44.
- If the specified authentication mode for the CAPF session is authentication-string, you must enter an authentication string on each phone that is receiving an updated LSC. For information, see Enter the Authentication String on the Phone, on page 47.
- To configure Media Encryption, see Configure Media Encryption (SRTP) in Cisco Unified CME, on page 51.

Configure Media Encryption (SRTP) in Cisco Unified CME

To configure the network for secure calls between Cisco Unified CME systems across an H.323 trunk, perform the following steps on the Cisco Unified CME router.



Restriction

- Secure three-way software conferencing is not supported. A secure call beginning with SRTP always falls back to nonsecure Real-Time Transport Protocol (RTP) when it is joined to a conference.
- If a party drops from a three-party conference, the call between the remaining two parties returns to secure if the two parties are SRTP-capable local Skinny Client Control Protocol (SCCP) endpoints to a single Cisco Unified CME and the conference creator is one of the remaining parties. If either of the two remaining parties are only RTP-capable, the call remains nonsecure. If the two remaining parties are connected through FXS, PSTN, or VoIP, the call remains nonsecure.
- Calls to Cisco Unity Express are not secure.
- Music on Hold (MOH) is not secure.
- Video calls are not secure.
- Modem relay and T.3 fax relay calls are not secure.
- Media flow-around is not supported for call transfer and call forward.
- Conversion between inband tone and RFC 2833 DTMF is not supported. RFC 2833 DTMF handling is supported when encryption keys are sent to secure DSP Farm devices but is not supported for codec passthrough.
- Secure Cisco Unified CME does not support SIP trunks; only H.323 trunks are supported.
- Media Encryption (SRTP) supports secure supplementary services in both H.450 and non-H.450 Cisco Unified CME networks. A secure Cisco Unified CME network should be either H.450 or non-H.450, not a hybrid.
- Secure calls are supported in the default session application only.

Before you begin

- Cisco Unified CME 4.2 or a later version.
- To make secure H.323 calls, telephony-service security parameters must be configured. See Configure Telephony-Service Security Parameters, on page 26.
- Compatible Cisco IOS Release on the Cisco VG224 Analog Phone Gateway. For information, see Cisco Unified CME and Cisco IOS Release Compatibility Matrix.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice service voip**
4. **supplementary-service media-renegotiate**
5. **srtp fallback**
6. **h323**
7. **emptycapability**
8. **exit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**voice service voip** **Example:** `Router(config)# voice service voip`	Enters voice-service configuration mode. • The **voip** keyword specifies VoIP encapsulation.
Step 4	**supplementary-service media-renegotiate** **Example:** `Router(conf-voi-serv)# supplementary-service media-renegotiate`	Enables midcall renegotiation of SRTP cryptographic keys.
Step 5	**srtp fallback** **Example:** `Router(conf-voi-serv)# srtp fallback`	Globally enables secure calls using SRTP for media encryption and authentication and enables SRTP-to-RTP fallback to support supplementary services such as ringback tone and MOH. • Skip this step if you are going to configure fallback on individual dial peers. • This command can also be configured in dial-peer configuration mode. This command in dial-peer configuration command takes precedence over this command in voice service voip configuration mode.
Step 6	**h323** **Example:** `Router(conf-voi-serv)# h323`	Enters H.323 voice-service configuration mode.
Step 7	**emptycapability** **Example:** `Router(conf-serv-h323)# emptycapability`	Eliminates the need for identical codec capabilities for all dial peers in the rotary group.
Step 8	**exit** **Example:** `Router(conf-serv-h323)# exit`	Exits H.323 voice-service configuration mode.

What to do next

You have completed the required task for configuring Media Encryption (SRTP) on Cisco Unified CME. Configuring Cisco Unified CME SRTP Fallback for H.323 Dial Peers. You can now perform the following optional tasks:

- Configure Cisco Unified CME SRTP Fallback for H.323 Dial Peers, on page 54(Optional)
- Configure Cisco Unity for Secure Cisco Unified CME Operation, on page 55(Optional)

Configure Cisco Unified CME SRTP Fallback for H.323 Dial Peers

To configure SRTP fallback for an individual dial peer, perform the following steps on the Cisco Unified CME router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **voice class codec** *tag*
4. **codec preference** *value codec-type*
5. **exit**
6. **dial-peer voice** *tag* **voip**
7. **srtp fallback**
8. **voice-class codec** *tag*
9. **exit**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**voice class codec** *tag* **Example:** `Router(config)# voice class codec 1`	Enters voice-class configuration mode and assigns an identification tag number for a codec voice class.
Step 4	**codec preference** *value codec-type* **Example:**	Specifies a list of preferred codecs to use on a dial peer. • Repeat this step to build a list of preferred codecs.

	Command or Action	Purpose
	`Router(config-voice-class)# codec preference 1 g711alaw`	• Use the same preference order for the codec list on both Cisco Unified CMEs on either side of the H.323 trunk.
Step 5	**exit** **Example:** `Router(config-voice-class)# exit`	Exits voice-class configuration mode.
Step 6	**dial-peer voice** *tag* **voip** **Example:** `Router(config)# dial-peer voice 101 voip`	Enters dial peer voice configuration mode.
Step 7	**srtp fallback** **Example:** `Router(config-dial-peer)# srtp fallback`	Enables secure calls that use SRTP for media encryption and authentication and specifies fallback capability. • Using the **no srtp** command disables SRTP and causes the dial peer to fall back to RTP mode. • **fallback**—Enables fallback to nonsecure mode (RTP) on an individual dial peer. The **no srtp fallback** command disables fallback and SRTP. • This command can also be configured in voice service voip configuration mode. This command in dial-peer configuration command takes precedence over this command in voice service voip configuration mode.
Step 8	**voice-class codec** *tag* **Example:** `Router(config-dial-peer)# voice-class codec 1`	Assigns a previously configured codec selection preference list (codec voice class) to a Voice over IP (VoIP) dial peer. • The *tag* argument in this step is the same as the *tag* in Step 3.
Step 9	**exit** **Example:** `Router(config-dial-peer)# exit`	Exits dial-peer voice configuration mode.

Configure Cisco Unity for Secure Cisco Unified CME Operation

This section contains the following tasks:

• Prerequisites for Configuring Cisco Unity for Secure Cisco Unified CME Operation, on page 56
• Configure Integration Between Cisco Unified CME and Cisco Unity, on page 56
• Import the Cisco Unity Root Certificate to Cisco Unified CME, on page 56
• Configure Cisco Unity Ports for Secure Registration, on page 58
• Verify that Cisco Unity are Registering Securely, on page 58

Prerequisites for Configuring Cisco Unity for Secure Cisco Unified CME Operation

- Cisco Unity 4.2 or later version.

Configure Integration Between Cisco Unified CME and Cisco Unity

To change the settings for the integration between Cisco Unified CME and Cisco Unity, perform the following steps on the Cisco Unity server:

Procedure

Step 1 If Cisco Unity Telephony Integration Manager (UTIM) is not yet open on the Cisco Unity server, choose**Programs** > **Cisco Unity** > **Manage Integrations** from the Windows Start menu. The UTIM window appears.

Step 2 In the left pane, double-click **Cisco Unity Server**. The existing integrations appear.

Step 3 Click **Cisco Unified Communications Manager** integration.

Step 4 In the right pane, click the cluster for the integration.

Step 5 Click the **Servers** tab.

Step 6 In the Cisco Unified Communications Manager Cluster Security Mode field, click the applicable setting.

Step 7 If you clicked **Non-secure**, click **Save** and skip the remaining steps in this procedure.

If you clicked **Authenticated** or **Encrypted**, the Security tab and the Add TFTP Server dialog box appear. In the IP Address or Host Name field of the Add TFTP Server dialog box, enter the IP address (or DNS name) of the primary TFTP server for the Cisco Unified Communications Manager cluster and click **OK**.

Step 8 If there are more TFTP servers that Cisco Unity will use to download the Cisco Unified Communications Manager certificates, click **Add**. The Add TFTP Server dialog box appears.

Step 9 In the IP Address or Host Name field, enter the IP address (or DNS name) of the secondary TFTP server for the Cisco Unified Communications Manager cluster and click **OK**.

Step 10 Click **Save**.

Cisco Unity creates the voice messaging port device certificates, exports the Cisco Unity server root certificate, and displays the Export Cisco Unity Root Certificate dialog box.

Step 11 Note the filename of the exported Cisco Unity server root certificate and click **OK**.

Step 12 On the Cisco Unity server, navigate to the CommServer\SkinnyCerts directory.

Step 13 Locate the Cisco Unity server root certificate file that you exported in Step 11.

Step 14 Right-click the file and click **Rename**.

Step 15 Change the file extension from .0 to .pem. For example, change the filename “12345.0” to “12345.pem” for the exported Cisco Unity server root certificate file.

Step 16 Copy this file to a PC from which you can access the Cisco Unified CME router.

Import the Cisco Unity Root Certificate to Cisco Unified CME

To import the Cisco Unity root certificate to Cisco Unified CME, perform the following steps on the Cisco Unified CME router:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint** *name*
4. **revocation-check none**
5. **enrollment terminal**
6. **exit**
7. **crypto pki authenticate** *trustpoint-label*
8. Open the root certificate file that you copied from the Cisco Unity Server in Step 16, on page 56.
9. You will be prompted to enter the CA certificate. Cut and paste the entire contents of the base 64 encoded certificate between BEGIN CERTIFICATE and END CERTIFICATE at the command line. Press **Enter** and type **quit**. The router prompts you to accept the certificate. Enter yes to accept the certificate.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**crypto pki trustpoint** *name* **Example:** `Router(config)# crypto pki trustpoint PEM`	Declares the trustpoint that your RA mode certificate server should use and enters ca-trustpoint configuration mode. • *label*—Name for the trustpoint and RA.
Step 4	**revocation-check none** **Example:** `Router(ca-trustpoint)# revocation-check none`	(Optional) Specifies that certificate checking is not required.
Step 5	**enrollment terminal** **Example:** `Router(ca-trustpoint)# enrollment terminal`	Specifies manual cut-and-paste certificate enrollment.
Step 6	**exit** **Example:** `Router(ca-trustpoint)# exit`	Exits ca-trustpoint configuration mode.
Step 7	**crypto pki authenticate** *trustpoint-label* **Example:**	Retrieves the CA certificate and authenticates it. Checks the certificate fingerprint when prompted.

	Command or Action	Purpose
	`Router(config)# crypto pki authenticate pem`	• *trustpoint-label*—Already-configured name for the trustpoint and RA. See Step 3, on page 57.
Step 8	Open the root certificate file that you copied from the Cisco Unity Server in Step 16, on page 56.	
Step 9	You will be prompted to enter the CA certificate. Cut and paste the entire contents of the base 64 encoded certificate between BEGIN CERTIFICATE and END CERTIFICATE at the command line. Press **Enter** and type **`quit`**. The router prompts you to accept the certificate. Enter yes to accept the certificate.	Completes the copying of the Cisco Unity root certificate to the Cisco Unified CME router.

Configure Cisco Unity Ports for Secure Registration

To configure Cisco Unity ports for registration in secure mode, perform the following steps:

Procedure

Step 1 Choose the Cisco voice-mail port that you want to update.

Step 2 From the Device Security Mode drop-down list, choose **Encrypted**.

Step 3 Click **Update**.

Verify that Cisco Unity are Registering Securely

Use the **show sccp connections** command to verify that Cisco Unity ports are registered securely with Cisco Unified CME.

In the following example, the secure value of the type field shows that the connections are secure.

```
Router# show sccp connections

  sess_id    conn_id    stype       mode     codec   ripaddr rport sport

  16777222   16777409   secure-xcode sendrecv g729b 10.3.56.120   16772 19534
  16777222   16777393   secure-xcode sendrecv g711u 10.3.56.50    17030 18464

  Total number of active session(s) 1, and connection(s) 2
```

HTTPS Provisioning for Cisco Unified IP Phones

To provision a Cisco Unified IP phone for secure access to web content using HTTPS, perform the following steps:

Before you begin

- Firmware 9.0 (4) or a later version must be installed on the IP phone to prevent an infinite registration loop.

- Certificate file to be imported from flash memory to the IP phone must be in privacy-enhanced mail format.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http server**
4. **crypto pki server** *cs-label*
5. **database level** {**minimum** | **names** | **complete**}
6. **database url** *root url*
7. **grant auto**
8. **exit**
9. **crypto pki trustpoint** *name*
10. **enrollment url** *url*
11. **exit**
12. **crypto pki server** *cs-label*
13. **no shutdown**
14. **exit**
15. **crypto pki trustpoint** *name*
16. **enrollment url** *url*
17. **revocation-check** *method1* [*method2* [*method3*]]
18. **rsakeypair** *key-label*
19. **exit**
20. **crypto pki authenticate** *name*
21. **crypto pki enroll** *name*
22. **crypto pki trustpoint** *name*
23. **enrollment url** *url*
24. **revocation-check** *method1* [*method2* [*method3*]]
25. **rsakeypair** *key-label*
26. **exit**
27. **crypto pki authenticate** *name*
28. **crypto pki enroll** *name*
29. **ctl-client**
30. **sastl trustpoint** *label*
31. **sast2 trustpoint** *label*
32. **import certificate** *tag description* **flash:** *cert_name*
33. **server application server address trustpoint label**
34. **regenerate**
35. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip http server** **Example:** `Router(config)# ip http server`	Enables the HTTP server on the Cisco Unified CME router.
Step 4	**crypto pki server** *cs-label* **Example:** `Router(config)# crypto pki server IOS-CA`	Enables a Cisco IOS certificate server and enters certificate server configuration mode. • *cs-label*—Name of the certificate server. **Note** The certificate server name should not exceed 13 characters.
Step 5	**database level** {**minimum** | **names** | **complete**} **Example:** `Router(cs-server)# database level complete`	Controls what type of data is stored in the certificate enrollment database. • **complete**—Each issued certificate is written to the database. If this keyword is used, you should enable the **database url** command.
Step 6	**database url** *root url* **Example:** `Router(cs-server)# database url flash:`	Specifies the location where database entries for the certificate server will be stored or published. • *root url*—Location where database entries will be written.
Step 7	**grant auto** **Example:** `Router(cs-server)# grant auto`	(Optional) Allows an automatic certificate to be issued to any requester. The recommended method and default if this command is not used is manual enrollment.
Step 8	**exit** **Example:** `Router(cs-server)# exit`	Exits certificate server configuration mode.

	Command or Action	Purpose
Step 9	**crypto pki trustpoint** *name* **Example:** `Router(config)# crypto pki trustpoint IOS-CA`	Declares a trustpoint and enters ca-trustpoint configuration mode. • *name*—Name for the trustpoint.
Step 10	**enrollment url** *url* **Example:** `Router(ca-trustpoint)# enrollment url http://10.1.1.1:80`	Specifies the enrollment parameters of a certification authority. • *url*—Specifies the URL of the file system where your router should send certificate requests.
Step 11	**exit** **Example:** `Router(ca-trustpoint)# exit`	Exits ca-trustpoint configuration mode.
Step 12	**crypto pki server** *cs-label* **Example:** `Router(config)# crypto pki server IOS-CA`	Enables a Cisco IOS certificate server and enters certificate server configuration mode. • *cs-label*—Name of the certificate server. **Note** The certificate server name should not exceed 13 characters.
Step 13	**no shutdown** **Example:** `Router(cs-server)# no shutdown`	Enables the Cisco IOS Certification Authority.
Step 14	**exit** **Example:** `Router(cs-server)# exit`	Exits certificate server configuration mode.
Step 15	**crypto pki trustpoint** *name* **Example:** `Router(config)# crypto pki trustpoint primary-cme`	Declares a trustpoint and enters ca-trustpoint configuration mode. • *name*—Name for the trustpoint.
Step 16	**enrollment url** *url* **Example:** `Router(ca-trustpoint)# enrollment url http://10.1.1.1:80`	Specifies the enrollment parameters of the certification authority. • *url*—Specifies the URL of the file system where your router should send certificate requests.
Step 17	**revocation-check** *method1* [*method2*[*method3*]] **Example:** `Router(ca-trustpoint)# revocation-check none`	Checks the revocation status of a certificate. • **none**—Certificate checking is not required.
Step 18	**rsakeypair** *key-label* **Example:**	Specifies which RSA key pair to associate with the certificate.

	Command or Action	Purpose
	`Router(ca-trustpoint)# rsakeypair primary-cme`	• *key-label*—Name of the key pair, which is generated during enrollment if it does not already exist or if the **auto-enroll regenerate** command is configured.
Step 19	**exit** **Example:** `Router(ca-trustpoint)# exit`	Exits ca-trustpoint configuration mode.
Step 20	**crypto pki authenticate** *name* **Example:** `Router(config)# crypto pki authenticate primary-cme`	Authenticates the certification authority by getting the authority's certificate. • *name*—Name of the certification authority.
Step 21	**crypto pki enroll** *name* **Example:** `Router(config)# crypto pki enroll primary-cme`	Obtains the certificates for the router from the certificate authority. • *name*—Name of the certification authority. Use the same name as when you declared the certification authority using the **crypto pki trustpoint** command.
Step 22	**crypto pki trustpoint** *name* **Example:** `Router(config)# crypto pki trustpoint sast-secondary`	Declares a trustpoint and enters ca-trustpoint configuration mode. • *name*—Name for the trustpoint.
Step 23	**enrollment url** *url* **Example:** `Router(ca-trustpoint)# enrollment url http://10.1.1.1:80`	Specifies the enrollment parameters of a certification authority. • *url*—Specifies the URL of the file system where your router should send certificate requests.
Step 24	**revocation-check** *method1* [*method2*[*method3*]] **Example:** `Router(ca-trustpoint)# revocation-check none`	Checks the revocation status of a certificate. • **none**—Certificate checking is not required.
Step 25	**rsakeypair** *key-label* **Example:** `Router(ca-trustpoint)# rsakeypair sast-secondary`	Specifies which RSA key pair to associate with the certificate. • *key-label*—Name of the key pair, which is generated during enrollment if it does not already exist or if the **auto-enroll regenerate** command is configured.
Step 26	**exit** **Example:** `Router(ca-trustpoint)# exit`	Exits ca-trustpoint configuration mode.
Step 27	**crypto pki authenticate** *name* **Example:**	Authenticates the certification authority by getting the authority's certificate.

	Command or Action	Purpose
	`Router(config)# crypto pki authenticate sast-secondary`	• *name*—Name of the certification authority.
Step 28	**crypto pki enroll** *name* **Example:** `Router(config)# crypto pki enroll sast-secondary`	Obtains the certificates for the router from the certificate authority. • *name*—Name of the certification authority. Use the same name as when you declared the certification authority using the **crypto pki trustpoint** command.
Step 29	**ctl-client** **Example:** `Router(config)# ctl-client`	Enters CTL-client configuration mode to set parameters for the CTL client.
Step 30	**sastl trustpoint** *label* **Example:** `Router(config-ctl-client)# sast1 trustpoint first-sast`	Configures the credentials for the primary SAST. • *label*—Name of SAST1 trustpoint. **Note** SAST1 and SAST2 certificates must be different from each other. The CTL file is always signed by SAST1. The SAST2 credentials are included in the CTL file so that if the SAST1 certificate is compromised, the file can be signed by SAST2 to prevent phones from being reset to the factory default.
Step 31	**sast2 trustpoint** *label* **Example:** `Router(config-ctl-client)# sast2 trustpoint second-sast`	Configures the credentials for the secondary SAST. • *label*—Name of SAST2 trustpoint. **Note** SAST1 and SAST2 certificates must be different from each other. The CTL file is always signed by SAST1. The SAST2 credentials are included in the CTL file so that if the SAST1 certificate is compromised, the file can be signed by SAST2 to prevent phones from being reset to the factory default.
Step 32	**import certificate** *tag description* **flash:** *cert_name* **Example:** `Router(config-ctl-client)# import certificate 5 FlashCert flash:flash_cert.cer`	Imports a trusted certificate in PEM format from flash memory to the CTL file of an IP phone. **Note** This step is required to provision HTTPS service running on external server. • *tag*—identifier for the trusted certificate. • *description*—Descriptive name of the trusted certificate. • **flash:***cert_cert*—Specifies the filename of the trusted certificate stored in flash memory.

	Command or Action	Purpose
Step 33	**server application server address trustpoint label** **Example:** `Router(config-ctl-client)# server application 10.1.2.3 trustpoint first-sast`	Configures the server application and the credentials for the SAST.
Step 34	**regenerate** **Example:** `Router(config-ctl-client)# regenerate`	Creates a new CTLFile.tlv after you make changes to the CTL client configuration.
Step 35	**end** **Example:** `Router(config-ctl-client)# end`	Exits to privileged EXEC mode.

Configuration Examples for Security

Example for Password and Key Removal from Logs

The following is a sample output for the show command, **show sip-ua calls**. The lines that are added to the show command output as part of the Unified CME 12.6 enhancement are the local crypto key and the remote crypto key.

```
SIP UAC CALL INFO
Number of SIP User Agent Client(UAC) calls: 0

SIP UAS CALL INFO
Call 1
SIP Call ID : 007278df-12e00376-6ed02377-6ffbaca9@8.55.0.195
State of the call : STATE_ACTIVE (7)
Substate of the call : SUBSTATE_NONE (0)
Calling Number : 1001
Called Number : 6901%23
Called URI : sip:6901%23@8.39.25.11;user=phone
Bit Flags : 0x10C0401C 0x10000100 0x4
CC Call ID : 196
Local UUID : 61488a9100105000a000007278df12e0
Remote UUID : c4b7f9475629538096ef61699b96746f
Source IP Address (Sig ): 8.39.25.11
Destn SIP Req Addr:Port : [8.55.0.195]:52704
Destn SIP Resp Addr:Port: [8.55.0.195]:52704
Destination Name : 8.55.0.195
Number of Media Streams : 1
Number of Active Streams: 1
RTP Fork Object : 0x0
Media Mode : flow-through
Media Stream 1
State of the stream : STREAM_ACTIVE
Stream Call ID : 196
Stream Type : voice+dtmf (1)
Stream Media Addr Type : 1
Negotiated Codec : g711ulaw (160 bytes)
```

```
Codec Payload Type : 0
Negotiated Dtmf-relay : rtp-nte
Dtmf-relay Payload Type : 101
QoS ID : -1
Local QoS Strength : BestEffort
Negotiated QoS Strength : BestEffort
Negotiated QoS Direction : None
Local QoS Status : None
Media Source IP Addr:Port: [8.39.25.11]:8080
Media Dest IP Addr:Port : [8.55.0.195]:23022
Local Crypto Suite : AEAD_AES_256_GCM
Remote Crypto Suite : AEAD_AES_256_GCM (
AEAD_AES_256_GCM
AEAD_AES_128_GCM
AES_CM_128_HMAC_SHA1_80
AES_CM_128_HMAC_SHA1_32 )
Local Crypto Key : 3taqc13ClF6BBpvd65WTMPrad/i0uyQ6iNouh+jYHxbf48d4TFmsOGyh4Vs=
Remote Crypto Key : 2/TNTV+Rc1Nh/wbGj0MGwIsLrJ4l+N2jKWGczolEnf7sgsA0Q9AEIz0a4eg=
Mid-Call Re-Assocation Count: 0
SRTP-RTP Re-Assocation DSP Query Count: 0
```

The following is a sample output for the show command, **show ephone offhook** . The lines that are added to the show command output as part of the Unified CME 12.6 enhancement are local key and remote key.

```
ephone-1[0] Mac:549A.EBB5.8000 TCP socket:[1] activeLine:1 whisperLine:0 REGISTERED in SCCP
 ver 21/17 max_streams=1 + Authentication + Encryption with TLS connection
mediaActive:1 whisper_mediaActive:0 startMedia:1 offhook:1 ringing:0 reset:0 reset_sent:0
paging 0 debug:0 caps:8
IP:8.44.22.63 * 17872 SCCP Gateway (AN) keepalive 28 max_line 1 available_line 1
port 0/0/0
button 1: cw:1 ccw:(0 0)
 dn 1 number 6901 CM Fallback CH1 CONNECTED CH2 IDLE
Preferred Codec: g711ulaw
Lpcor Type: none Active Secure Call on DN 1 chan 1 :6901 8.44.22.63 18116
 to 8.39.25.11 8066 via 8.39.0.1
G711Ulaw64k 160 bytes no vad
SRTP cipher: AES_CM_128_HMAC_SHA1_32
  local key: 0OPV0yxvcnRLPMzHfmYbwgHfdxcuS1uPbp5j/Tjk
  remote key: e8DQl3Kvk7LjZlipaCoMg9TMreBmiPsFmNiVHwIA
Tx Pkts 0 bytes 0 Rx Pkts 0 bytes 0 Lost 0
Jitter 0 Latency 0 callingDn -1 calledDn -1
```

Example for Configuring Unified CME for Password Policy

The following is a sample configuration on Unified CME router to support password encryption:

```
Router(config)#key config-key password-encrypt <cisco123>
Router(config)#password encryption aes
Router(config)#telephony-service
Router(config-telephony)encrypt password
```



Note Configure **no encrypt password** for password unencryption (type 0) on the Unified CME router. If type 0 is configured, the password is displayed as unencrypted plain text.

Example for Configuring Cisco IOS CA

```
crypto pki server iosca
   grant auto
   database url flash:
  !
  crypto pki trustpoint iosca
   revocation-check none
   rsakeypair iosca
  !
  crypto pki certificate chain iosca
   certificate ca 01
    308201F9 30820162 ...
```

Example for Manually Importing MIC Root Certificate on the Cisco Unified CME Router

The following example shows three certificates imported to the router (7970, 7960, PEM):

```
Router(config)# crypto pki trustpoint 7970
  Router(ca-trustpoint)# revocation-check none
  Router(ca-trustpoint)# enrollment terminal
  Router(ca-trustpoint)# exit
  Router(config)# crypto pki authenticate 7970

  Enter the base 64 encoded CA certificate.
  End with a blank line or the word "quit" on a line by itself
  MIIDqDCCApCgAwIBAgIQNT+yS9cPFKNGwfOprHJWdTANBgkqhkiG9w0BAQUFADAu
  MRYwFAYDVQQKEw1DaXNjbyBTeXN0ZW1zMRQwEgYDVQQDEwtDQVAtUlRQLTAwMjAe
  Fw0wMzEwMTAyMDE4NDlaFw0yMzEwMTAyMDI3MzdaMC4xFjAUBgNVBAoTDUNpc2Nv
  IFN5c3RlbXMxFDASBgNVBAMTC0NBUC1SVFAtMDAyMIIBIDANBgkqhkiG9w0BAQEF
  AAOCAQ0AMIIBCAKCAQEAxCZlBK19w/2NZVVvpjCPrpW1cCY7V1q9lhzI85RZZdnQ
  2M4CufgIzNa3zYxGJIAYeFfcRECnMB3f5A+x7xNiEuzE87UPvK+7S80uWCY0Uhtl
  AVVf5NQgZ3YDNoNXg5MmONb8lT86F55EZyVac0XGne77TSIbidejrTgYQXGP2MJx
  Qhg+ZQlGFDRzbHfM84Duv2Msez+l+SqmqO80kIckqE9Nr3/XCSj1hXZNNVg8D+mv
  Hth2P6KZqAKXAAStGRLSZX3jNbS8tveJ3Gi5+sj9+F6KKK2PD0iDwHcRKkcUHb7g
  lI++U/5nswjUDIAph715Ds2rn9ehkMGipGLF8kpuCwIBA6OBwzCBwDALBgNVHQ8E
  BAMCAYYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUUpIr4ojuLgmKTn5wLFal
  mrTUm5YwbwYDVR0fBGgwZjBkoGKgYIYtaHR0cDovL2NhcC1ydHAtMDAyL0NlcnRF
  bnJvbGwvQ0FQLVJUUC0wMDIuY3Jshi9maWxlOi8vXFxjYXAtcnRwLTAwMlxDZXJ0
  RW5yb2xsXENBUC1SVFAtMDAyLmNybDAQBgkrBgEEAYI3FQEEAwIBADANBgkqhkiG
  9w0BAQUFAAOCAQEAVoOM78TaOtHqj7sVL/5u5VChlyvU168f0piJLNWip2vDRihm
  E+DlXdwMS5JaqUtuaSd/m/xzxpcRJm4ZRRwPq6VeaiiQGkjFuZEe5jSKiSAK7eHg
  tup4HP/ZfKSwPA40DlsGSYsKNMm3OmVOCQUMH02lPkS/eEQ9sIw6QS7uuHN4y4CJ
  NPnRbpFRLw06hnStCZHtGpKEHnY213QOy3h/EWhbnp0MZ+hdr20FujSI6G1+L39l
  aRjeD708f2fYoz9wnEpZbtn2Kzse3uhU1Ygq1D1x9yuPq388C18HWdmCj4OVTXux
  V6Y47H1yv/GJM8FvdgvKlExbGTFnlHpPiaG9tQ==
  quit
  Certificate has the following attributes:
  Fingerprint MD5: F7E150EA 5E6E3AC5 615FC696 66415C9F
  Fingerprint SHA1: 1BE2B503 DC72EE28 0C0F6B18 798236D8 D3B18BE6
  % Do you accept this certificate? [yes/no]: y
  Trustpoint CA certificate accepted.
  % Certificate successfully imported
  Router(config)# crypto pki trustpoint 7960
  Router(ca-trustpoint)# revocation-check none
  Router(ca-trustpoint)# enrollment terminal
  Router(ca-trustpoint)# exit
```

```
Router(config)# crypto pki authenticate 7960

Enter the base 64 encoded CA certificate.

End with a blank line or the word "quit" on a line by itself
MIICKDCCAZGgAwIBAgIC8wEwDQYJKoZIhvcNAQEFBQAwQDELMAkGA1UEBhMCVVMx
GjAYBgNVBAoTEUNpc2NvIFN5c3RlbXMgSW5jMRUwEwYDVQQDEwxDQVBGLTdEN0Qw
QzAwHhcNMDQwNzE1MjIzODMyWhcNMTkwNzEyMjIzODMxWjBAMQswCQYDVQQGEwJV
UzEaMBgGA1UEChMRQ2lzY28gU3lzdGVtcyBJbmMxFTATBgNVBAMTDENBUEYtN0Q3
RDBDMDCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA0hvMOZZ9ENYWme11YGY1
it2rvE3Nk/eqhnv8P9eqB1iqt+fFBeAG0WZ5bO5FetdU+BCmPnddvAeSpsfr3Z+h
x+r58fOEIBRHQLgnDZ+nwYH39uwXcRWWqWwlW147YHjV7M5c/R8T6daCx4B5NBo6
kdQdQNOrV3IP7kQaCShdM/kCAwEAAaMxMC8wDgYDVR0PAQH/BAQDAgKEMB0GA1Ud
JQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDBTANBgkqhkiG9w0BAQUFAAOBgQCaNi6x
sL6M5NlDezpSBO3QmUVyXMfrONV2ysrSwcXzHu0gJ9MSJ8TwiQmVaJ47hSTlF5a8
YVYJ0idifXbXRo+/EEO7kkmFE8MZta5rM7UWj8bAeR42iqA3RzQaDwuJgNWT9Fhh
GgfuNAlo5h1Aikxsvxi vmDlLdZyCMoqJJd7B2Q==
quit
Certificate has the following attributes:
Fingerprint MD5: 4B9636DF 0F3BA6B7 5F54BE72 24762DBC
Fingerprint SHA1: A9917775 F86BB37A 5C130ED2 3E528BB8 286E8C2D
% Do you accept this certificate? [yes/no]: y
Trustpoint CA certificate accepted.
% Certificate successfully imported


Router(config)# crypto pki trustpoint PEM
Router(ca-trustpoint)# revocation-check none
Router(ca-trustpoint)# enrollment terminal
Router(ca-trustpoint)# exit
Router(config)# crypto pki authenticate PEM

Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself
MIIDqDCCApCgAwIBAgIQdhL5YBU9b59OQiAgMrcjVjANBgkqhkiG9w0BAQUFADAu
MRYwFAYDVQQKEw1DaXNjbyBTeXN0ZW1zMRQwEgYDVQQDEwtDQVAtUlRQLTAwMTAe
Fw0wMzAyMDYyMzI3MTNaFw0yMzAyMDYyMzM2MzRaMC4xFjAUBgNVBAoTDUNpc2Nv
IFN5c3RlbXMxFDASBgNVBAMTC0NBUC1SVFAtMDAxMIIBIDANBgkqhkiG9w0BAQEF
AAOCAQ0AMIIBCAKCAQEArFW77Rjem4cJ/7yPLVCauDohwZZ/3qf0sJaWlLeAzBlq
Rj2lFlSij0ddkDtfEEo9VKmBOJsvx6xJlWJiuBwUMDhTRbsuJz+npkaGBXPOXJmN >
Vd54qlpc/hQDfWlbrIFkCcYhHws7vwnPsLuy1Kw2L2cP0UXxYghSsx8H4vGqdPFQ
NnYy7aKJ43SvDFt4zn37n8jrvlRuz0x3mdbcBEdHbA825Yo7a8sk12tshMJ/YdMm
vny0pmDNZXmeHjqEgVO3UFUn6GVCO+K1y1dUU1qpYJNYtqLkqj7wgccGjsHdHr3a
U+bw1uLgSGsQnxMWeMaWo8+6hMxwlANPweufgZMaywIBA6OBwzCBwDALBgNVHQ8E
BAMCAYYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQU6Rexgscfz6ypG270qSac
cK4FoJowbwYDVR0fBGgwZjBkoGKgYIYtaHR0cDovL2NhcC1ydHAtMDAxL0NlcnRF
bnJvbGwvQ0FQLVJUUC0wMDEuY3Jshi9maWxlOi8vXFxjYXAtcnRwLTAwMVxDZXJ0
RW5yb2xsXENBUC1SVFAtMDAxLmNybDAQBgkrBgEEAYI3FQEEAwIBADANBgkqhkiG
9w0BAQUFAAOCAQEAq2T96/YMMtw2Dw4QX+F1+g1XSrUCrNyjx7vtFaRDHyB+kobw
dwkpohfkzfTyYpJELzV1r+kMRoyuZ7oIqqccEroMDnnmeApc+BRGbDJqS1Zzk4OA
c6Ea7fm53nQRlcSPmUVLjDBzKYDNbnEjizptaIC5fgB/S9S6C1q0YpTZFn5tjUjy
WXzeYSXPrcxb0UH7IQJ1ogpONAAUKLoPaZU7tVDSH3hD4+VjmLyysaLUhksGFrrN
phzZrsVVilK17qpqCPllKLGAS4fSbkruq3r/6S/SpXS6/gAoljBKixP7ZW2PxgCU
1aU9cURLPO95NDOFN3jBk3Sips7cVidcogowPQ==
quit

Certificate has the following attributes:
Fingerprint MD5: 233C8E33 8632EA4E 76D79FEB FFB061C6
Fingerprint SHA1: F7B40B94 5831D2AB 447AB8F2 25990732 227631BE
% Do you accept this certificate? [yes/no]: y
Trustpoint CA certificate accepted.
```

```
% Certificate successfully imported
```

Use the **show crypto pki trustpoint status** command to show that enrollment has succeeded and that five CA certificates have been granted. The five certificates include the three certificates just entered, the CA server certificate, and the router certificate.

```
Router# show crypto pki trustpoint status

  Trustpoint 7970:
  Issuing CA certificate configured:
  Subject Name:
  cn=CAP-RTP-002,o=Cisco Systems
  Fingerprint MD5: F7E150EA 5E6E3AC5 615FC696 66415C9F
  Fingerprint SHA1: 1BE2B503 DC72EE28 0C0F6B18 798236D8 D3B18BE6
  State:
  Keys generated ..... Yes (General Purpose)
  Issuing CA authenticated ..... Yes
  Certificate request(s) ..... None
  Trustpoint 7960:
  Issuing CA certificate configured:
  Subject Name:
  cn=CAPF-508A3754,o=Cisco Systems Inc,c=US
  Fingerprint MD5: 6BAE18C2 0BCE391E DAE2FE4C 5810F576
  Fingerprint SHA1: B7735A2E 3A5C274F C311D7F1 3BE89942 355102DE
  State:
  Keys generated ..... Yes (General Purpose)
  Issuing CA authenticated ..... Yes
  Certificate request(s) ..... None
  Trustpoint PEM:
  Issuing CA certificate configured:
  Subject Name:
  cn=CAP-RTP-001,o=Cisco Systems
  Fingerprint MD5: 233C8E33 8632EA4E 76D79FEB FFB061C6
  Fingerprint SHA1: F7B40B94 5831D2AB 447AB8F2 25990732 227631BE
  State:
  Keys generated ..... Yes (General Purpose)
  Issuing CA authenticated ..... Yes
  Certificate request(s) ..... None
  Trustpoint srstcaserver:
  Issuing CA certificate configured:
  Subject Name:
  cn=srstcaserver
  Fingerprint MD5: 6AF5B084 79C93F2B 76CC8FE6 8781AF5E
  Fingerprint SHA1: 47D30503 38FF1524 711448B4 9763FAF6 3A8E7DCF
  State:
  Keys generated ..... Yes (General Purpose)
  Issuing CA authenticated ..... Yes
  Certificate request(s) ..... None

  Trustpoint srstca:
  Issuing CA certificate configured:
  Subject Name:
  cn=srstcaserver
  Fingerprint MD5: 6AF5B084 79C93F2B 76CC8FE6 8781AF5E
  Fingerprint SHA1: 47D30503 38FF1524 711448B4 9763FAF6 3A8E7DCF
  Router General Purpose certificate configured:
  Subject Name:
  serialNumber=F3246544+hostname=c2611XM-sSRST.cisco.com
  Fingerprint: 35471295 1C907EC1 45B347BC 7A9C4B86
  State:
  Keys generated ..... Yes (General Purpose)
  Issuing CA authenticated ..... Yes
  Certificate request(s) ..... Yes
```

Example for Configuring Telephony-Service Security Parameters

The following example shows Cisco Unified CME security parameters:

```
 telephony-service
  device-security-mode authenticated
  secure-signaling trustpoint cme-sccp
  tftp-server-credentials trustpoint cme-tftp
  load-cfg-file slot0:Ringlist.xml alias Ringlist.xml sign create

 ephone 24
  device-security-mode authenticated
  capf-auth-str 2734
  cert-oper upgrade auth-mode auth-string
```

Example for Configuring CTL Client Running on Cisco Unified CME Router

```
ctl-client
 server capf 10.1.1.1 trustpoint cmeserver
 server cme 10.1.1.1 trustpoint cmeserver
 server tftp 10.1.1.1 trustpoint cmeserver
 sast1 trustpoint cmeserver
 sast2 trustpoint sast2 CTL Client Running on Another Router: Example
ctl-client
 server cme 10.1.1.100 trustpoint cmeserver
 server cme 10.1.1.1 username cisco password 1 0822455D0A16544541
 sast1 trustpoint cmeserver
 sast2 trustpoint sast1 CAPF Server: Example
!
ip dhcp pool cme-pool
   network 10.1.1.0 255.255.255.0
   option 150 ip 10.1.1.1
   default-router 10.1.1.1
!
capf-server
 port 3804
 auth-mode null-string
 cert-enroll-trustpoint iosra password 1 00071A1507545A545C
 trustpoint-label cmeserver
 source-addr 10.1.1.1
!
crypto pki server iosra
 grant auto
 mode ra
 database url slot0:
!
crypto pki trustpoint cmeserver
 enrollment url http://10.1.1.100:80
 serial-number
 revocation-check none
 rsakeypair cmeserver
!
crypto pki trustpoint sast2
 enrollment url http://10.1.1.100:80
 serial-number
 revocation-check none
 rsakeypair sast2
!
!

crypto pki trustpoint iosra
   enrollment url http://10.1.1.200:80
```

```
 revocation-check none
 rsakeypair iosra
!
!
crypto pki certificate chain cmeserver
 certificate 1B
  30820207 30820170 A0030201 0202011B 300D0609 2A864886 F70D0101 04050030
  ....
  quit
 certificate ca 01
  3082026B 308201D4 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  ...
  quit
crypto pki certificate chain sast2
 certificate 1C
  30820207 30820170 A0030201 0202011C 300D0609 2A864886 F70D0101 04050030
  ....
  quit
 certificate ca 01
  3082026B 308201D4 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  .....
  quit
crypto pki certificate chain capf-tp
crypto pki certificate chain iosra
 certificate 04
  30820201 3082016A A0030201 02020104 300D0609 2A864886 F70D0101 04050030
  ......
 certificate ca 01
  308201F9 30820162 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  ....
  quit
!
!
credentials
 ctl-service admin cisco secret 1 094F471A1A0A464058
 ip source-address 10.1.1.1 port 2444
 trustpoint cmeserver
!
!
telephony-service
 no auto-reg-ephone
 load 7960-7940 P00307010200
 load 7914 S00104000100
 load 7941GE TERM41.7-0-0-129DEV
 load 7970 TERM70.7-0-0-77DEV
 max-ephones 20
 max-dn 10
 ip source-address 10.1.1.1 port 2000 secondary 10.1.1.100
 secure-signaling trustpoint cmeserver
 cnf-file location flash:
 cnf-file perphone
 dialplan-pattern 1 2... extension-length 4
 max-conferences 8 gain -6
 transfer-pattern ....
 tftp-server-credentials trustpoint cmeserver
 server-security-mode secure
 device-security-mode encrypted
 load-cfg-file slot0:Ringlist.xml alias Ringlist.xml sign
 load-cfg-file slot0:P00307010200.bin alias P00307010200.bin
 load-cfg-file slot0:P00307010200.loads alias P00307010200.loads
 load-cfg-file slot0:P00307010200.sb2 alias P00307010200.sb2
 load-cfg-file slot0:P00307010200.sbn alias P00307010200.sbn
 load-cfg-file slot0:cnu41.2-7-4-116dev.sbn alias cnu41.2-7-4-116dev.sbn
 load-cfg-file slot0:Jar41.2-9-0-101dev.sbn alias Jar41.2-9-0-101dev.sbn
```

```
 load-cfg-file slot0:CVM41.2-0-0-96dev.sbn alias CVM41.2-0-0-96dev.sbn
 load-cfg-file slot0:TERM41.DEFAULT.loads alias TERM41.DEFAULT.loads
 load-cfg-file slot0:TERM70.DEFAULT.loads alias TERM70.DEFAULT.loads
 load-cfg-file slot0:Jar70.2-9-0-54dev.sbn alias Jar70.2-9-0-54dev.sbn
 load-cfg-file slot0:cnu70.2-7-4-58dev.sbn alias cnu70.2-7-4-58dev.sbn
 load-cfg-file slot0:CVM70.2-0-0-49dev.sbn alias CVM70.2-0-0-49dev.sbn
 load-cfg-file slot0:DistinctiveRingList.xml alias DistinctiveRingList.xml sign
 load-cfg-file slot0:Piano1.raw alias Piano1.raw sign
 load-cfg-file slot0:S00104000100.sbn alias S00104000100.sbn
 create cnf-files version-stamp 7960 Aug 13 2005 12:39:24
!
!
ephone  1
 device-security-mode encrypted
 cert-oper upgrade auth-mode null-string
 mac-address 000C.CE3A.817C
 type 7960 addon 1 7914
 button  1:2 8:8
!
!
ephone  2
 device-security-mode encrypted
 capf-auth-str 2476
 cert-oper upgrade auth-mode null-string
 mac-address 0011.2111.6BDD
 type 7970
 button  1:1
!
!
ephone  3
 device-security-mode encrypted
 capf-auth-str 5425
 cert-oper upgrade auth-mode null-string
 mac-address 000D.299D.50DF
 type 7970
 button  1:3
!
!
ephone  4
 device-security-mode encrypted
 capf-auth-str 7176
 cert-oper upgrade auth-mode null-string
 mac-address 000E.D7B1.0DAC
 type 7960
 button  1:4
!
!
ephone  5
 device-security-mode encrypted
 mac-address 000F.9048.5077
 type 7960
 button  1:5
!
!
ephone  6
 device-security-mode encrypted
 mac-address 0013.C352.E7F1
 type 7941GE
 button  1:6
!
```

Example for Secure Unified CME

```
Router# show running-config

  Building configuration...

  Current configuration : 12735 bytes
  !
  ! No configuration change since last restart
  !
  version 12.4
  service timestamps debug datetime msec
  service timestamps log datetime msec
  no service password-encryption
  service internal
  !
  hostname Router
  !
  boot-start-marker
  boot-end-marker
  !
  card type e1 1 1
  logging queue-limit 1000
  logging buffered 9999999 debugging
  logging rate-limit 10000
  no logging console
  !
  aaa new-model
  !
  !
  aaa accounting connection h323 start-stop group radius
  !
  aaa session-id common
  !
  resource policy
  !
  clock timezone IST 5
  no network-clock-participate slot 1
  !
  !
  ip cef
  !
  !
  isdn switch-type primary-net5
  !
  voice-card 0
   no dspfarm
  !
  voice-card 1
   no dspfarm
  !
  !
  ctl-client
   server capf 10.13.32.11 trustpoint mytrustpoint1
   server tftp 10.13.32.11 trustpoint mytrustpoint1
   server cme 10.13.32.11 trustpoint mytrustpoint1
   sast1 trustpoint mytrustpoint1>
   sast2 trustpoint sast2
  !
  capf-server
   port 3804
   auth-mode null-string
   cert-enroll-trustpoint iosra password 1 mypassword
```

```
 trustpoint-label mytrustpoint1
 source-addr 10.13.32.11
 phone-key-size 512
!
voice call debug full-guid
!
voice service voip
 srtp fallback
 allow-connections h323 to h323
 no supplementary-service h450.2
 no supplementary-service h450.3
 no supplementary-service h450.7
 supplementary-service media-renegotiate
 h323
  emptycapability
  ras rrq ttl 4000
!
!
voice class codec 2
 codec preference 1 g711alaw
 codec preference 2 g711ulaw
!
voice class codec 3
 codec preference 1 g729r8
 codec preference 8 g711alaw
 codec preference 9 g711ulaw
!
voice class codec 1
 codec preference 1 g729r8
 codec preference 2 g728
 codec preference 3 g723ar63
 codec preference 4 g711ulaw
!
!
voice iec syslog
voice statistics type iec
voice statistics time-range since-reset
!
!
!
crypto pki server myra
 database level complete
 grant auto
 lifetime certificate 1800
!
crypto pki trustpoint myra
 enrollment url http://10.13.32.11:80
 revocation-check none
 rsakeypair iosra
!
crypto pki trustpoint mytrustpoint1
 enrollment url http://10.13.32.11:80
 revocation-check none
 rsakeypair mytrustpoint1
!
crypto pki trustpoint sast2
 enrollment url http://10.13.32.11:80
 revocation-check none
 rsakeypair sast2
!
!
crypto pki certificate chain myra
 certificate ca 01
  308201F9 30820162 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
```

```
  10310E30 0C060355 04031305 696F7372 61301E17 0D303630 37303730 35343031
  375A170D 30393037 30363035 34303137 5A301031 0E300C06 03550403 1305696F
  73726130 819F300D 06092A86 4886F70D 01010105 0003818D 00308189 02818100
  D8CE29F9 C9FDB1DD 0E1517E3 6CB4AAF7 52B83DE2 1C017ACA DFC4AF42 F9D10D08
  E74BF95B 29378902 B49E32C4 85907384 84CAE4B2 7759BB84 8AB1F578 580793C4
 B11A2DBE B2ED02CC DA0C3824 A5FCC377 18CE87EA C0C297BA BE54530F E62247D8
  1483CD14 9FD89EFE 05DFBB37 E03FD3F8 B2B1C0B8 A1931BCC B1174A9E 6566F8F5
  02030100 01A36330 61300F06 03551D13 0101FF04 05300301 01FF300E 0603551D
  0F0101FF 04040302 0186301F 0603551D 23041830 168014B7 16F6FD67 29666C90
  D0C62515 E14265A9 EB256230 1D060355 1D0E0416 0414B716 F6FD6729 666C90D0
  C62515E1 4265A9EB 2562300D 06092A86 4886F70D 01010405 00038181 002B7F41
  64535A66 D20D888E 661B9584 5E3A28DF 4E5A95B9 97E57CAE B07A7C38 7F3B60EE
  75C7E5DE 6DF19B06 5F755FB5 190BABFC EF272CEF 865FE01B 1CE80F98 F320A569
  CAFFA5D9 3DB3E7D8 8A86C66C F227FF81 6C4449F2 AF8015D9 8129C909 81AFDC01
  180B61E8 85E19873 96DB3AE3 E6B70726 9BF93521 CA2FA906 99194ECA 8F
  quit
crypto pki certificate chain mytrustpoint1
 certificate 02
  308201AB 30820114 A0030201 02020102 300D0609 2A864886 F70D0101 04050030
  10310E30 0C060355 04031305 696F7372 61301E17 0D303630 37303730 35343233
 385A170D 30393037 30363035 34303137 5A301A31 18301606 092A8648 86F70D01
  09021609 32383531 2D434D45 32305C30 0D06092A 864886F7 0D010101 0500034B
  00304802 4100B3ED A902646C 3851B7F6 CF94887F 0EC437E3 3B6FEDB2 2B4B45A6
  3611C243 5A0759EA 1E8D96D1 60ABE028 ED6A3F2A E95DCE45 BE0921AF 82E53E57
  17CC12F0 C1270203 010001A3 4F304D30 0B060355 1D0F0404 030205A0 301F0603
  551D2304 18301680 14B716F6 FD672966 6C90D0C6 2515E142 65A9EB25 62301D06
  03551D0E 04160414 4EE1943C EA817A9E 7010D5B8 0467E9B0 6BA76746 300D0609
  2A864886 F70D0101 04050003 81810003 564A6DA1 868B2669 7C096F9A 41173CFC
  E49246EE C645E30B A0753E3B E1A265D1 6EA5A829 F10CD0E8 3F2E3AD4 39D8DFE8
  83525F2B D19F5E15 F27D6262 62852D1F 43629B68 86D91B5F 7B2E2C25 3BD2CCC3
  00EF4028 714339B2 6A7E0B2F 131D2D9E 0BE08853 5CCAE47C 4F74953C 19305A20
  B2C97808 D6E01351 48366421 A1D407
  quit
 certificate ca 01
  308201F9 30820162 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  10310E30 0C060355 04031305 696F7372 61301E17 0D303630 37303730 35343031
  375A170D 30393037 30363035 34303137 5A301031 0E300C06 03550403 1305696F
  73726130 819F300D 06092A86 4886F70D 01010105 0003818D 00308189 02818100
  D8CE29F9 C9FDB1DD 0E1517E3 6CB4AAF7 52B83DE2 1C017ACA DFC4AF42 F9D10D08
  E74BF95B 29378902 B49E32C4 85907384 84CAE4B2 7759BB84 8AB1F578 580793C4
  B11A2DBE B2ED02CC DA0C3824 A5FCC377 18CE87EA C0C297BA BE54530F E62247D8
  1483CD14 9FD89EFE 05DFBB37 E03FD3F8 B2B1C0B8 A1931BCC B1174A9E 6566F8F5
  02030100 01A36330 61300F06 03551D13 0101FF04 05300301 01FF300E 0603551D
  0F0101FF 04040302 0186301F 0603551D 23041830 168014B7 16F6FD67 29666C90
  D0C62515 E14265A9 EB256230 1D060355 1D0E0416 0414B716 F6FD6729 666C90D0
  C62515E1 4265A9EB 2562300D 06092A86 4886F70D 01010405 00038181 002B7F41
  64535A66 D20D888E 661B9584 5E3A28DF 4E5A95B9 97E57CAE B07A7C38 7F3B60EE
  75C7E5DE 6DF19B06 5F755FB5 190BABFC EF272CEF 865FE01B 1CE80F98 F320A569
  CAFFA5D9 3DB3E7D8 8A86C66C F227FF81 6C4449F2 AF8015D9 8129C909 81AFDC01
  180B61E8 85E19873 96DB3AE3 E6B70726 9BF93521 CA2FA906 99194ECA 8F
  quit
crypto pki certificate chain sast2
 certificate 03
  308201AB 30820114 A0030201 02020103 300D0609 2A864886 F70D0101 04050030
  10310E30 0C060355 04031305 696F7372 61301E17 0D303630 37303730 35343331
  375A170D 30393037 30363035 34303137 5A301A31 18301606 092A8648 86F70D01
  09021609 32383531 2D434D45 32305C30 0D06092A 864886F7 0D010101 0500034B
  00304802 4100C703 840B11A7 81FCE5AE A14FE593 5114D3C2 5473F488 B8FB4CC5
  41EAFA3A D99381D8 21AE6AA9 BA83A84E 9DF3E8C6 54978787 5EF6CC35 C334D55E
  A3051372 17D30203 010001A3 4F304D30 0B060355 1D0F0404 030205A0 301F0603
  551D2304 18301680 14B716F6 FD672966 6C90D0C6 2515E142 65A9EB25 62301D06
  03551D0E 04160414 EB2146B4 EE24AA61 8B5D2F8D 2AD3B786 CBADC8F2 300D0609
  2A864886 F70D0101 04050003 81810057 BA0053E9 8FD54B25 72D85A4C CAB47F26
  8316F494 E94DFFB9 8E9D065C 9748465C F54719CA C7724F50 67FBCAFF BC332109
```

```
  DC2FB93D 5AD86583 EDC3E648 39274CE8 D4A5F002 5F21ED3C 6D524AB7 7F5B1876
  51867027 9BD2FFED 06984558 C903064E 5552015F 289BA9BB 308D327A DFE0A3B9
  78CF2B02 2DD4C208 80CDC0A8 43A26A
  quit
 certificate ca 01
  308201F9 30820162 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  10310E30 0C060355 04031305 696F7372 61301E17 0D303630 37303730 35343031
  375A170D 30393037 30363035 34303137 5A301031 0E300C06 03550403 1305696F
  73726130 819F300D 06092A86 4886F70D 01010105 0003818D 00308189 02818100
  D8CE29F9 C9FDB1DD 0E1517E3 6CB4AAF7 52B83DE2 1C017ACA DFC4AF42 F9D10D08
  E74BF95B 29378902 B49E32C4 85907384 84CAE4B2 7759BB84 8AB1F578 580793C4
  B11A2DBE B2ED02CC DA0C3824 A5FCC377 18CE87EA C0C297BA BE54530F E62247D8
  1483CD14 9FD89EFE 05DFBB37 E03FD3F8 B2B1C0B8 A1931BCC B1174A9E 6566F8F5
  02030100 01A36330 61300F06 03551D13 0101FF04 05300301 01FF300E 0603551D
  0F0101FF 04040302 0186301F 0603551D 23041830 168014B7 16F6FD67 29666C90
  D0C62515 E14265A9 EB256230 1D060355 1D0E0416 0414B716 F6FD6729 666C90D0
  C62515E1 4265A9EB 2562300D 06092A86 4886F70D 01010405 00038181 002B7F41
  64535A66 D20D888E 661B9584 5E3A28DF 4E5A95B9 97E57CAE B07A7C38 7F3B60EE
  75C7E5DE 6DF19B06 5F755FB5 190BABFC EF272CEF 865FE01B 1CE80F98 F320A569
  CAFFA5D9 3DB3E7D8 8A86C66C F227FF81 6C4449F2 AF8015D9 8129C909 81AFDC01
  180B61E8 85E19873 96DB3AE3 E6B70726 9BF93521 CA2FA906 99194ECA 8F
  quit
!
!
username admin password 0 mypassword2
username cisco password 0 mypassword2
!
!
controller E1 1/0
 pri-group timeslots 1-31
!
controller E1 1/1
 pri-group timeslots 1-31
gw-accounting aaa
!
!
!
!
!
interface GigabitEthernet0/0
 ip address 10.13.32.11 255.255.255.0
 duplex auto
 speed auto
 fair-queue 64 256 32
 h323-gateway voip interface
 h323-gateway voip id GK1 ipaddr 10.13.32.13 1719
 h323-gateway voip id GK2 ipaddr 10.13.32.16 1719
 h323-gateway voip h323-id 2851-CiscoUnifiedCME
 h323-gateway voip tech-prefix 1#
 ip rsvp bandwidth 1000 100
!
interface GigabitEthernet0/1
 no ip address
 shutdown
 duplex auto
 speed auto
!
interface Serial1/0:15
 no ip address
 encapsulation hdlc
 isdn switch-type primary-net5
 isdn protocol-emulate network
 isdn incoming-voice voice
 no cdp enable
```

```
!
interface Serial1/1:15
 no ip address
 encapsulation hdlc
 isdn switch-type primary-net5
 isdn protocol-emulate network
 isdn incoming-voice voice
 no cdp enable
!
ip route 0.0.0.0 0.0.0.0 10.13.32.1
!
!
ip http server
ip http authentication local
no ip http secure-server
ip http path flash:
!
!
!
!
!
!
tftp-server flash:music-on-hold.au
tftp-server flash:TERM70.DEFAULT.loads
tftp-server flash:TERM71.DEFAULT.loads
tftp-server flash:P00308000300.bin
tftp-server flash:P00308000300.loads
tftp-server flash:P00308000300.sb2
tftp-server flash:P00308000300.sbn
tftp-server flash:SCCP70.8-0-3S.loads
tftp-server flash:cvm70sccp.8-0-2-25.sbn
tftp-server flash:apps70.1-1-2-26.sbn
tftp-server flash:dsp70.1-1-2-26.sbn
tftp-server flash:cnu70.3-1-2-26.sbn
tftp-server flash:jar70sccp.8-0-2-25.sbn
radius-server host 10.13.32.241 auth-port 1645 acct-port 1646
radius-server timeout 40
radius-server deadtime 2
radius-server key cisco
radius-server vsa send accounting
!
control-plane
!
no call rsvp-sync
!
!
voice-port 1/0/0
!
voice-port 1/0/1
!
voice-port 1/0:15
!
voice-port 1/1:15
!
!
!
!
!
dial-peer voice 1 voip
 destination-pattern ........
 voice-class codec 2
 session target ras
 incoming called-number 9362....
 dtmf-relay h245-alphanumeric
```

```
 req-qos controlled-load audio
!
dial-peer voice 2 pots
 destination-pattern 93621101
!
dial-peer voice 3 pots
 destination-pattern 93621102
!
dial-peer voice 10 voip
 destination-pattern 2668....
 voice-class codec 1
 session target ipv4:10.13.46.200
!
dial-peer voice 101 voip
 shutdown
 destination-pattern 5694....
 voice-class codec 1
 session target ipv4:10.13.32.10
 incoming called-number 9362....
!
dial-peer voice 102 voip
 shutdown
 destination-pattern 2558....
 voice-class codec 1
 session target ipv4:10.13.32.12
 incoming called-number 9362....
!
dial-peer voice 103 voip
 shutdown
 destination-pattern 9845....
 voice-class codec 1
 session target ipv4:10.13.32.14
 incoming called-number 9362....
!
dial-peer voice 104 voip
 shutdown
 destination-pattern 9844....
 voice-class codec 1
 session target ipv4:10.13.32.15
 incoming called-number 9362....
!
dial-peer voice 201 pots
 destination-pattern 93625...
 no digit-strip
 direct-inward-dial
 port 1/0:15
!
dial-peer voice 202 pots
 destination-pattern 93625...
 no digit-strip
 direct-inward-dial
 port 1/1:15
!
!
gateway
 timer receive-rtp 1200
!
!
!
telephony-service
 load 7960-7940 P00308000300
 max-ephones 4
 max-dn 4
 ip source-address 10.13.32.11 port 2000
```

```
 auto assign 1 to 4
 secure-signaling trustpoint mytrustpoint1
 cnf-file location flash:
 cnf-file perphone
 voicemail 25589000
 max-conferences 4 gain -6
call-forward pattern .T
 moh flash:music-on-hold.au
 web admin system name admin password mypassword2
 dn-webedit
 time-webedit
 transfer-system full-consult
 transfer-pattern ........
 tftp-server-credentials trustpoint mytrustpoint1
 server-security-mode secure
 device-security-mode encrypted
 create cnf-files version-stamp 7960 Oct 25 2006 07:19:39
!
!
ephone-dn  1
 number 93621000
 name 2851-PH1
 call-forward noan 25581101 timeout 10
!
!
ephone-dn  2
 number 93621001
 name 2851-PH2
 call-forward noan 98441000 timeout 10
!
!
ephone-dn  3
 number 93621002
 name 2851-PH3
!
!
ephone-dn  4
 number 93621003
 name 2851-PH4
!
!
ephone  1
       capf-ip-in-cnf
             no multicast-moh
 device-security-mode encrypted
 mac-address 0012.4302.A7CC
 type 7970
 button  1:1
!
!
!
ephone  2
             capf-ip-in-cnf
 no multicast-moh
 device-security-mode encrypted
 mac-address 0017.94CA.9CCD
 type 7960
 button  1:2
!
!
!
ephone  3
             capf-ip-in-cnf
 no multicast-moh
```

```
 device-security-mode encrypted
 mac-address 0017.94CA.9833
 type 7960
 button  1:3
!
!
!
ephone  4
             capf-ip-in-cnf
 no multicast-moh
 device-security-mode none
 mac-address 0017.94CA.A141
 type 7960
 button  1:4
!
!
!
line con 0
 logging synchronous level all limit 20480000
line aux 0
line vty 0 4
!
scheduler allocate 20000 1000
ntp clock-period 17179791
ntp server 10.13.32.12
!
webvpn context Default_context
 ssl authenticate verify all
 !
 no inservice
!
!
end
```

Example for Configuring HTTPS Support for Cisco Unified CME

Configurations similar to the following example are required before HTTPS support for services like local-directory lookup, My Phone Apps, and Extension Mobility in Cisco Unified CME can be configured at four different levels:

```
Router(config)# ip http server
Router(config)# crypto pki server IOS-CA
Router(cs-server)# database level complete
Router(cs-server)# database url flash:
Router(cs-server)# grant auto
Router(cs-server)# exit
Router(config)# crypto pki trustpoint IOS-CA
Router(ca-trustpoint)# enrollment url http://10.1.1.1:80
Router(ca-trustpoint)# exit
Router(config)# crypto pki server IOS-CA
Router(cs-server)# no shutdown
Router(cs-server)# exit
Router(config)# crypto pki trustpoint primary-cme
Router(ca-trustpoint)# enrollment url http://10.1.1.1.80
Router(ca-trustpoint)# revocation-check none
Router(ca-trustpoint)# rsakeypair primary-cme
Router(ca-trustpoint)# exit
Router(config)# crypto pki authenticate primary-cme
Router(config)# crypto pki enroll primary-cme
Router(config)# crypto pki trustpoint sast-secondary
Router(ca-trustpoint)# enrollment url http://10.1.1.1:80
Router(ca-trustpoint)# revocation-check none
```

```
Router(ca-trustpoint)# rsakeypair sast-secondary
Router(ca-trustpoint)# exit
Router(config)# crypto pki authenticate sast-secondary
Router(config)# crypto pki enroll sast-secondary
Router(config)# ctl-client
Router(config-ctl-client)# sast1 trustpoint first-sast
Router(config-ctl-client)# sast2 trustpoint second-sast
Router(config-ctl-client)# server application 10.1.2.3 trustpoint first-sast
Router(config-ctl-client)# regenerate
Router(config-ctl-client)# end
```

For Cisco Unified SCCP IP Phones at the global level:

```
configure terminal
telephony-service
     cnf-file perphone
     service https
```

For Cisco Unified SCCP IP Phones at the ephone-template level:

```
configure terminal
ephone-template 1
     service https
```

For Cisco Unified SIP IP Phones at the global level:

```
configure terminal
voice register global
     service https
```

For Cisco Unified SIP IP Phones at the voice register template level:

```
configure terminal
voice register template 1
    service https
```

Where to Go Next

PKI Management

Cisco IOS public key infrastructure (PKI) provides certificate management to support security protocols such as IP Security (IPsec), secure shell (SSH), and secure socket layer (SSL).

Cisco VG224 Analog Phone Gateway

- To configure secure endpoints on the Cisco VG224 Analog Phone Gateway, see the *Configuring Secure Signalling and Media Encryption on the Cisco VG224* section of Supplementary Services Features for FXS Ports on Cisco IOS Voice Gateways Configuration Guide.

Feature Information for Security

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to https://cfnng.cisco.com/. An account on Cisco.com is not required.

Table 3: Feature Information for Security

Feature Name	Cisco Unified CME Version	Feature Information
Unified CME Password Policy	12.6	Introduces password policy enforcement for Unified CME
HTTPS Support in Cisco Unified CME	9.5	Introduces HTTPS support on Cisco Unified CME.
HTTPS Provisioning for Cisco Unified IP Phones	8.8	Allows you to import an IP phone's trusted certificate to an IP phone's CTL file using the **import certificate** command.
Media Encryption (SRTP) on Cisco Unified CME	4.2	Introduces media encryption on Cisco Unified CME.
Phone Authentication	4.0	Introduces phone authentication for Cisco Unified CME phones.