Configuration Example for Secure SIP Integration Between CUCM and CUC based on Next Generation Encryption (NGE)

Contents

Introduction
Prerequisites
Requirements
Network Diagram
Certificate requirements
RSA key based ciphers negotiated
EC Key Based ciphers negotiated
Configure - Cisco Unity Connection (CUC)
1. Add a New Port Group
2. Add the TFTP server reference
3. Add Voice Mail Ports
4. Upload CUCM Root and intermediate certificate of the third party CA
Configure - Cisco Unified CM (CUC)
1. Create a SIP trunk security profile
2. Create a secure SIP Trunk
3. Configure TLS and SRTP ciphers
4. Upload CUC Tomcat certificates (RSA & EC based)
5. Create Route pattern
6. Create Voicemail Pilot, Voicemail Profile and assign it to the DNs
Configure - Signing the EC key based certificates by third party CA (optional)
Verify
Secure SIP Trunk verification
Secure RTP Call Verification
Related Information

Introduction

This document describes the configuration and verification of the secure SIP connection between the Cisco Unified Communication Manager (CUCM) and Cisco Unity Connection (CUC) server using Next Generation Encryption.

Next Generation Security over SIP interface restricts SIP interface to use Suite B ciphers based on TLS 1.2, SHA-2 and AES256 protocols. It allows the various combinations of ciphers based on the priority order of RSA or ECDSA ciphers. During the communication between Unity Connection and Cisco Unified CM, both ciphers and third party certificates are verified at both the ends. Below is the configuration for Next Generation Encryption support.
If you plan to use the certificates signed by third party Certification Authority then start with Certificate signing at the end of the configuration section (Configure - Signing the EC key based certificates by third party CA)

Prerequisites

Requirements

The information in this document is based on these software and hardware versions:

CUCM version 11.0 and later in Mixed mode
CUC version 11.0 and later

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Network Diagram

This diagram briefly explains the process that helps establish a secure connection between CUCM and CUC once the Next generation encryption support is enabled:
Certificate requirements

These are the certificate exchange requirements once the Next generation encryption support is enabled on Cisco Unity Connection.

- RSA key based ciphers negotiated
  - CUCM certificate used
    - CallManager.pem (self-signed)
    - CallManager.pem (CA signed)
  - CUC certificate used
    - Tomcat.pem (self-signed)
    - Tomcat.pem (CA signed)
  - Certs to upload to CUCM
    - Tomcat.pem to be uploaded into CUCM > CallManager-trust
      - CUC root & intermediate CA certificate\(^1\) to be uploaded into CUCM > CallManager-trust
  - Certs to upload to CUC
    - None.
      - CUCM root & intermediate certificate\(^2\) to be uploaded into CUC > CallManager-trust.

\(^1\) CUCM root & intermediate certificate
\(^2\) CUCM root & intermediate certificate to be uploaded into CUC > CallManager-trust

CUC root & intermediate CA certificate refers to CA certificate which signed the Unity connection Tomcat certificate (Tomcat.pem).

CUCM root & intermediate CA certificate refers to CA certificate which signed the CUCM CallManager certificate (Callmanager.pem).

### EC Key Based ciphers negotiated

<table>
<thead>
<tr>
<th>CUCM certificate used</th>
<th>CUC certificate used</th>
<th>Certs to upload to CUCM</th>
<th>Certs to upload to CUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>CallManager-ECDSA.pem (self-signed)</td>
<td>Tomcat-ECDSA.pem (self-signed)</td>
<td>Tomcat-ECDSA.pem to be uploaded into CUCM &gt; CallManager-trust</td>
<td>None.</td>
</tr>
<tr>
<td>CallManager-ECDSA.pem (CA signed)</td>
<td>Tomcat-ECDSA.pem (CA signed)</td>
<td>CallManager-trust.</td>
<td>CUCM root &amp; intermediate CA certificate(^1) to be uploaded into CUCM &gt; CallManager-trust.</td>
</tr>
<tr>
<td>CallManager-ECDSA.pem (CA signed)</td>
<td>Tomcat-ECDSA.pem (self-signed)</td>
<td>CallManager-trust.</td>
<td>CUCM root &amp; intermediate CA certificate(^2) to be uploaded into CUC &gt; CallManager-trust.</td>
</tr>
<tr>
<td>CallManager-ECDSA.pem (self-signed)</td>
<td>Tomcat-ECDSA.pem (CA signed)</td>
<td>CUCM root &amp; intermediate CA certificate to be uploaded into CUCM &gt; CallManager-trust.</td>
<td>None.</td>
</tr>
</tbody>
</table>

\(^1\) CUC root & intermediate CA certificate refers to CA certificate which signed the Unity connection EC based Tomcat certificate (Tomcat-ECDSA.pem).

\(^2\) CUCM root & intermediate CA certificate refers to CA certificate which signed the CUCM CallManager certificate (CallManager-ECDSA.pem).

1. **Note**: Tomcat-ECDSA.pem certificate is called CallManager-ECDSA.pem in 11.0.1 versions of CUC. From CUC 11.5.x the certificate has been renamed to Tomcat-ECDSA.pem.

## Configure - Cisco Unity Connection (CUC)

### 1. Add a New Port Group

Navigate to Cisco Unity Connection Administration page > Telephony integration > Port group and Click on Add New. Make sure to check the Enable Next Generation Encryption checkbox.
Note: Unity Connection's Cisco Tomcat certificate will be used during SSL handshake once the Enable Next Generation Encryption checkbox is enabled.

- In case ECDSA based cipher is negotiated then EC key based tomcat-ECDSA certificate is used in SSL handshake.
- In case RSA based cipher is negotiated then RSA key based tomcat certificate is used in SSL handshake.

2. Add the TFTP server reference

On the Port Group Basics page, navigate to Edit > Servers and add FQDN of TFTP server of your CUCM cluster. FQDN/Hostname of the TFTP server must match the Common name (CN) of CallManager certificate. IP address of the server will not work and it will result in failure to download the ITL file. The DNS name must be therefore resolvable via configured DNS server.
Restart Connection Conversation Manager on each node by navigating to Cisco Unity Connection Serviceability > Tools > Service Management. This is mandatory for the configuration to take effect.

1. **Note:** Unity connection downloads ITL file (ITLfile.tlv) from CUCM's TFTP using https protocol on secure 6972 port (URL: https://<CUCM-TFTP-FQDN>:6972/ITLFile.tlv). CUCM must be in the mixed mode since CUC is looking for the "CCM+TFTP" function certificate from the ITL file.

Navigate back to Telephony integration > Port group > Port Group Basics configuration page and reset your newly added Port group.

1. **Note:** Every time the port group is reset, the CUC server will update its locally stored ITL file by connecting to CUCM server.

### 3. Add Voice Mail Ports

Navigate back to Telephony integration > Port and click on Add new to add port to your newly created port group.
4. Upload CUCM Root and intermediate certificate of the third party CA

In case of third party certificates, you must upload the Root and Intermediate certificate of the third party Certification Authority on CallManager-trust of Unity Connection. This is needed only if 3rd party CA signed your Call Manager certificate. Perform this action by navigating to Cisco Unified OS Administration > Security > Certificate Management and click on Upload Certificate.

Configure - Cisco Unified CM (CUCM)

1. Create a SIP trunk security profile

Navigate to CUCM Administration > System > Security > SIP Trunk Security Profile and add a new profile. X.509 Subject Name must match the FQDN of the CUC server.
1. **Note**: CLI command "show cert own tomcat/tomcat.pem" can display the RSA key based tomcat certificate on Unity Connection. It's CN must match the X.509 Subject name configured on CUCM. The CN is equal to FQDN/Hostname of the Unity server. The EC key based certificate contains the FQDN/hostname in its Subject Alternate Name (SAN) field.

2. **Create a secure SIP Trunk**

Navigate to Device > Trunk > Click and Add new and create a standard SIP trunk which will be used for secure integration with Unity Connection.
3. Configure TLS and SRTP ciphers

1. **Note:** The negotiation between Unity Connection and Cisco Unified Communications Manager depends on the TLS cipher configuration with the following conditions: When Unity Connection acts as server, TLS cipher negotiation is based on the preference selected by Cisco Unified CM. In case ECDSA based cipher is negotiated then EC key based tomcat-ECDSA certificates are used in SSL handshake. In case RSA based cipher is negotiated then RSA key based tomcat certificates are used in SSL handshake. When Unity Connection acts as client, TLS cipher negotiation is based on the preference selected by Unity Connection. Navigate to Cisco Unified CM > Systems > Enterprise Parameters and select the appropriate
cipher option from the TLS and SRTP Ciphers from drop-down list.

Restart the Cisco Call Manager service on each node by navigating to Cisco Unified Serviceability page, Tools > Control Centre-Feature Services and select Cisco Call Manager under CM Services

Navigate to Cisco Unity Connection Administration page > System Settings > General Configurations and select the appropriate cipher option from the TLS and SRTP Ciphers from drop-down list.

TLS Cipher options with Priority order

**TLS Cipher Options**

- **Strongest- AES-256 SHA-384 Only: RSA Preferred**
- **Strongest-AES-256 SHA-384 Only: ECDSA Preferred**
- **Medium-AES-256 AES-128 Only: RSA Preferred**

**TLS Ciphers in Priority Order**

- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_RSA_WITH_AES_256_CBC_SHA

**Medium-AES-256 AES-128 Only: ECDSA Preferred**

- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_RSA_WITH_AES_128_CBC_SHA

**All Ciphers RSA Preferred (Default)**

- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_RSA_WITH_AES_128_CBC_SHA

**All Ciphers ECDSA Preferred**

- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_RSA_WITH_AES_128_CBC_SHA

**SRTP Cipher Options in Priority order**

<table>
<thead>
<tr>
<th>SRTP Cipher Option</th>
<th>SRTP in Priority Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>All supported AES-256, AES-128 ciphers</td>
<td>AEAD_AES_256_GCM</td>
</tr>
<tr>
<td></td>
<td>AEAD_AES_128_GCM</td>
</tr>
<tr>
<td></td>
<td>AES_CM_128_HMAC_SHA1_32</td>
</tr>
<tr>
<td>AEAD AES-256, AES-28 GCM-based ciphers</td>
<td>AEAD_AES_256_GCM</td>
</tr>
<tr>
<td>AEAD AES256 GCM-based ciphers only</td>
<td>AEAD_AES_128_GCM</td>
</tr>
<tr>
<td></td>
<td>AEAD_AES_256_GCM</td>
</tr>
</tbody>
</table>

**4. Upload CUC Tomcat certificates (RSA & EC based)**

Navigate to OS Administration > Security > Certificate Management and upload both CUC Tomcat certificates (RSA & EC based) into the CallManager-trust store.
1. **Note**: Uploading both Unity Tomcat certificates is not mandatory if ECDSA ciphers are negotiated only. In such case EC based Tomcat certificate is enough. In case of third party certificates, you must upload the root and Intermediate certificate of the third party Certification Authority. This is needed only if 3rd party CA signed your Unity Tomcat certificate.

Restart the Cisco Call Manager process on all nodes to apply the changes.

5. **Create Route pattern**

Configure a route pattern that points to the configured trunk by navigating to Call Routing > Route/Hunt > Route Pattern. Extension entered as a route pattern number can be used as a voicemail pilot.

6. **Create Voicemail Pilot, Voicemail Profile and assign it to the DNs**

Create a voice mail pilot for the integration by going to Advanced Features > Voice Mail > Voice
Create a voice mail profile in order to link all the settings together Advanced Features > Voice Mail > Voice Mail Profile

Assign the newly created voice mail profile to the DNs intended to use the secure integration by going to Call Routing > Directory number

Configure - Signing the EC key based certificates by third party CA (optional)

The certificates might be signed by a third party CA before setting up the secure integration between the systems. Follow the following steps to sign the certificates on both systems.

Cisco Unity Connection

1. Generate Certificate signing request (CSR) for CUC Tomcat-ECDSA and have the certificate signed by third party CA
2. CA provides Identity certificate (CA signed certificate) and CA certificate (CA root certificate) which must be uploaded as follow:
   - Upload CA root certificate into the tomcat-trust store
   - Upload Identity certificate into the tomcat-EDCS store
3. Restart Conversation Manager on CUC

Cisco Unified CM

1. Generate CSR for CUCM CallManager-ECDSA and have the certificate signed by third party CA
2. CA provides Identity certificate (CA signed certificate) and CA certificate (CA root certificate) which must be uploaded as follow:
   - Upload CA root certificate into the callmanager-trust store
   - Upload Identity certificate into the callmanager-EDCS store

3. Restart Cisco CCM and TFTP services on each node

The same process will be used to sign RSA key based certificates where CSR is generated for CUC Tomcat certificate and CallManager certificate and uploaded into the tomcat store and callmanager store respectively.

**Verify**

Use this section to confirm that your configuration works properly.

**Secure SIP Trunk verification**

Press the Voice Mail button on the phone to call voice mail. You should hear the opening greeting if the user's extension is not configured on the Unity Connection system.

Alternatively, you can enable SIP OPTIONs keepalive to monitor the SIP trunk status. This option can be enabled in the SIP profile assigned to the SIP trunk. Once this is enabled you can monitor the Sip trunk status via Device > Trunk as shown below:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Calling Search Space</th>
<th>Device Pool</th>
<th>Route Pattern</th>
<th>Trunk Type</th>
<th>SIP Trunk Status</th>
<th>SIP Trunk Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUCv11</td>
<td></td>
<td>Default</td>
<td></td>
<td>2000</td>
<td>SIP Trunk</td>
<td>Full Service</td>
<td>Three in Full Service: 0 day 0 hour 0 minute</td>
</tr>
</tbody>
</table>

**Secure RTP Call Verification**

Verify whether the padlock icon is present on calls to Unity Connection. It means RTP stream is encrypted (Device Security profile must be secure in order for it to work) as shown in this image.
Related Information

- SIP Integration Guide for Cisco Unity Connection Release 11.x