ICE Telephony Administration Guide

Product guide for prerelease

Document version 1841, produced on Friday, September 06, 2024.

main 90adc8bf40040649230176bbdd465f6261a2d8e0

Contents

List of Tables

1 Document History

Publication Date	Product Release	Notes
May 28, 2024	3.5.1	Updated version reference to 3.5.7732.
April 15, 2024	3.5.0	Updated version reference to 3.5.7682.
October 27, 2023	3.4.0	Removed the *'Installing local patch server on Ubuntu via docker'* and *'Installing local static reflector on Ubuntu via docker'* sections and moved them to the **'ICE Server Installation Guide'**.
October 23, 2023	3.4.0	Updated version reference to 3.4.7353.
September 20, 2023	3.4.0	Updated version to 3.4.7252.
July 24, 2023	3.3.0	Updated version reference to 3.3.7007. Updated Docker run command.
April 25, 2023	3.2.0	Updated ICE Telephony (gateway) version references to 3.2.6822. Added three variables to the `env` file.
December 31, 2022	3.2.0	Leaned up some code formatting.
December 1, 2022	3.2.0	Updated ICE Telephony (gateway) version references to 3.2.6425.
September 26, 2022	3.1.2	Updated ICE Telephony (gateway) version references to 3.1.6084.
August 24, 2022	3.1.1	Replaced the term 'engagebridge' with the term 'patch' in most instances.
June 9, 2022	3.1.1	Updated ICE Telephony (gateway) version references to 3.1.5521.
May 31, 2022	3.1.1	Updated to point SIP Trunk to ICE Telephony Server port 5060 (was 7070). Updated command to see TCP/UDP port utilization.
April 25, 2022	3.1.1	Multiple updates for release.
March 15, 2022	3.1.0	Document created.

2 Introduction

ICE Telephony integrates Instant Connect Enterprise's push-to-talk communications with your SIP PBX as registrar or as SIP Trunk, enabling advanced voice communication features, like:

- A telephone caller can dial an Instant Connect user (using ICE Desktop or ICE Android) and establish a full-duplex phone call with them.
- An appropriately configured Instant Connect user can use their client software to place a dial call. In this regard, the ICE Desktop and ICE Android clients function as a "soft phone."
- A telephone caller can dial directly into a channel that's been configured to accept outside callers. The telephone caller can speak on the channel by pressing the * key to request the floor, and the # to relinquish it.

3 Prerequisites

Assure that each of the following prerequisites have been met before proceeding with installation.

3.1 Cisco Unified Communications Manager 11.5 or 12.5 or equivalent SIP Registrar

ICE Telephony works by configuring a custom SIP Trunk or as third party SIP advanced endpoints and registering dial numbers(DNs) with a SIP registrar and has been tested to work with Cisco Unified Communications Manager (CUCM) versions 11.5 and 12.5.

These instructions assume the reader is familiar with configuring end users and associated DNs on their CUCM or equivalent registrar. Assignment of dial numbers on the registrar will be required to complete the installation.

3.2 Host System

ICE Telephony has been designed to run outside of the ICE Server's Kubernetes cluster. This deployment model allows network administrators to maintain separate networks for their telephone system and their clustered applications.

The ICE Telephony software acts as a bridge between these systems, picking up SIP/telephony traffic from the network and distributing it to Instant Connect users via a RallyPoint. It should be deployed on a host system on a network with full network access to telephony communications.

3.2.1 Hardware requirements

ICE Telephony should be installed on a physical server or virtual machine that can dedicate the following resources to it:

- Ubuntu Linux 18.04 LTS or 20.04 LTS (Server and non-desktop version)
- 4 CPU cores (or equivalent)
- 4 GB RAM
- 80 GB storage

3.2.2 Software requirements

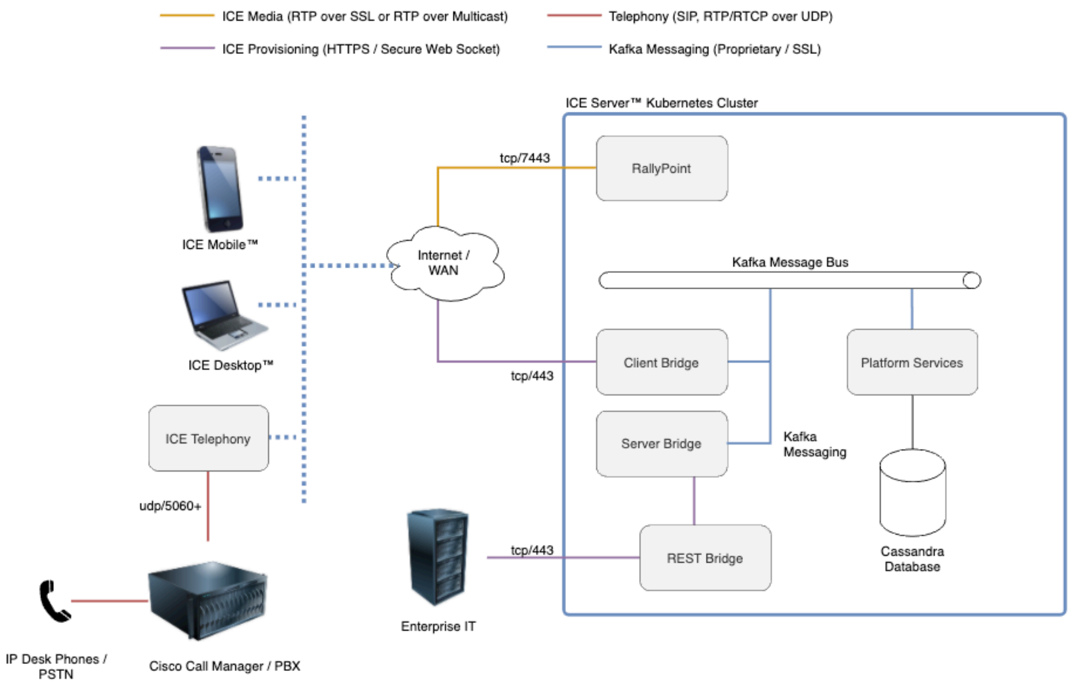
ICE Telephony is delivered as a Docker container that is intended to run on a Linux host operating system. Aside from Docker, ICE Telephony requires no additional software to be present on the host system.

At this time, only Linux host systems are supported.

3.2.3 Network requirements

The ICE Telephony component requires connectivity to two primary systems:

- The ICE Server. ICE Telephony will connect to ICE Server in the same manner as an ICE Desktop or ICE Mobile client would (using either an HTTP or HTTPS web socket connection).
- The RallyPoint configured as the default telephony RallyPoint, and to any other RallyPoint that may be used by a channel that has been configured to allow dial-in users.



3.3 Feature License

ICE Telephony is an independently licensed feature of Instant Connect Enterprise. Assure that your system has this feature license installed before proceeding. To do so: Launch ICE Desktop, connect to your ICE Server, navigate to Settings/License and look for the presence of a `VOICE_PORT_CHANNELS` feature in the "Licensed Features" panel. If you do not see this feature, contact your Instant Connect sales representative for assistance.

4 Setup

The installation of ICE Telephony is intended to occur after your ICE Server system has been installed, licensed and tested. Complete the primary system configuration before proceeding with telephony integration.

4.1 Install Docker

Follow the official Docker installation instructions for your Linux distribution if your system is not already equipped with Docker.

For Ubuntu: Install docker using steps in https://docs.docker.com/engine/install/ubuntu/

For RedHat 7:

```
sudo yum install docker -y
sudo systemctl enable docker.service
sudo systemctl start docker.service
sudo usermod -a -G dockerroot $(whoami)
sudo setfacl --modify user:$(whoami):rw /var/run/docker.sock
```

Verify the user ID (UID) and group ID (GID) specify 1000. If the IDs specify another user, like 1001, then Docker won't mount the logs to the `telephony-logs` folder.

To determine what user the IDs currently specify, enter:

```
id
```

If the output is not…

```
uid=1000(iceadmin) gid=1000(iceadmin) groups=1000(iceadmin)
```

…then run the following commands:

- To specify UID = 1000, enter:

```
sudo usermod -u 1000 iceadmin
```

- To specify GID = 1000, enter:

```
sudo groupmod -g 1000 iceadmin
```

Log out from putty/ssh or reboot the VM, so that the user has permissions to access the Docker daemon.

Verify your Docker installation with `docker run hello-world` which should pull the `hello-world` Docker image and run it, displaying some information about your installation.

> **Note:** Do *not* proceed until you have verified your Docker environment is able to run the `hello-world` test.

4.2 Connect to the Instant Connect Docker repository

You'll also need to log into the Instant Connect private Docker repository to pull and run our Docker container.

Log into Docker with the following command:

```
docker \
  login docker.io \
  -u iccustomeraccess \
  -p 7dcb7799-f418-4651-ab65-66feec2a4234
```

Then, verify you can pull the ICE Telephony container by executing

```
docker pull instantconnect/gateway:3.5.7732
```

> **Note:** For now, you will have to manually enter the IP addresses of the Arcus cluster you are connecting to and the public IP address of the node you're using to host ICE-Telephony into the file `env`

We need to pass in several environment variables, we do so by creating an `env` file to feed into docker. It's easiest to build this from your workstation, then upload it to your telephony node. The structure of the file is as follows:

```
ICE_TEL_PLATFORM_URL=https://chicago.icnow.app
CLIENT_BRIDGE_ADDRESS=https://chicago.icnow.app
SERVER_BRIDGE_ADDRESS=https://chicago.icnow.app/server-bridge
INGRESS_IP=your telephony node ip
GATEWAY_PLATFORM_LOGIN_TOKEN=<token from your ICE Server>
GATEWAY_TYPE=telephony
ICEGW_ARCUS_RELOGIN_ATTEMPTS_COUNT=10
ICEGW_ARCUS_WAIT_INTERVAL_BETWEEN_RELOGIN_ATTEMPTS=60
ICEGW_ARCUS_MAX_ATTEMPTS_TO_DOWNLOAD_CONFIG=10
```

> **Note:** The ICE Server environment variables must include `https://` or `http://` at the beginning, but the `INGRESS_IP` variable **cannot** include it. For server without certificates or FQDN, use http://<Your Server IP>

You need to populate `GATEWAY_PLATFORM_LOGIN_TOKEN` based on a Kubernetes secrets for the ICE Telephony to register to ICE Server. If you have kubectl access on your ICE Server cluster, you can obtain the token value (`jq 1.6` or newer is required):

1. Run the following command:

   ```
   kubectl -n ice-arcus get secrets gateway-auth-token -o json
   ```

2. Copy the `sip` value from the output, e.g., if the output were this...

   ```
   "data": {"dfsig":
   "bWh4ZnVmNWlwMHJueTYwcGdxdmJwbTA1NWt5MjNxYmdnaXIxejJwM2JoaTMyanp0aXc
   0bGQwNTE1c3M5NHI0Zg==","sip":
   "aGJ3enF2ZmwxdGc4Nm01a2FjNnA2Z3F5MTM1NHBtdGhoYmhiNXdxYmtyZW54YjkzZjd
   2NmdrMXlraG43azd2aQ=="},
   ```

…then you would copy the `sip` value:

```
aGJ3enF2ZmwxdGc4Nm01a2FjNnA2Z3F5MTM1NHBtdGhoYmhiNXdxYmtyZW54YjkzZjd2Nm
drMXlraG43azd2aQ==
```

3. Using the above example, you then run the following command:

```
echo
    aGJ3enF2ZmwxdGc4Nm01a2FjNnA2Z3F5MTM1NHBtdGhoYmhiNXdxYmtyZW54YjkzZjd

2NmdrMXlraG43azd2aQ== | base64 -d
```

4. The output is the token value, e.g.,:

```
hbwzqvfl1tg86m5kac6p6gqy1354pmthhbhb5wqbkrenxb93f7v6gk1ykhn7k7vi
```

5. Copy the token value and use it for `GATEWAY_PLATFORM_LOGIN_TOKEN` in `env`. In our example, the `env` file on the ICE Telephony node would have following information:

```
ICE_TEL_PLATFORM_URL=https://chicago.icnow.app
CLIENT_BRIDGE_ADDRESS=https://chicago.icnow.app
SERVER_BRIDGE_ADDRESS=https://chicago.icnow.app/server-bridge
INGRESS_IP=your telephony node ip
GATEWAY_PLATFORM_LOGIN_TOKEN=
    hbwzqvfl1tg86m5kac6p6gqy1354pmthhbhb5wqbkrenxb93f7v6gk1ykhn7k7vi
GATEWAY_TYPE=telephony
```

> **Notes:**
>
> - If your cluster is not using certificate, replace `https` with `http`.
> - The token must not include double-quotation marks.

5 Running the container

To run the image from your Telephony host node, use the following command:

> We recommend you create the `env` file and run the following commands in the home directory (simply type `cd` to get there).

```
mkdir -p telephony-logs
docker run  --detach \
--net=host \
--volume $(pwd)/certs:/usr/local/share/ca-certificates \
--name telephony \
--env-file env \
--restart always \
```

```
-v $(pwd)/telephony-logs:/home/gateway/ice/logs \
--log-driver json-file \
--log-opt max-size=1g \
instantconnect/gateway:3.5.7732 \
&& docker exec -it telephony update-ca-certificates
```

Useful commands for viewing the status are:

```
docker container ls -a
docker ps -a
docker stats
```

If you need to restart and/or stop and remove the telephony container:

```
docker restart telephony && docker stop telephony && docker rm telephony
```

> **Important:** Removing container with `docker rm telephony` would not reclaim disk space. Docker images can consume large amounts of disk space. Consult the official Docker Docs website to learn how to remove unused Docker images with the `docker image prune -a` command. `docker system prune -a` will also clean up unused containers and images but exercise caution in deleting constainers that are in use.

To view logs on console:

```
docker logs telephony
```

To see TCP/UDP port utilization:

```
netstat -anp | grep ice-gw
```

6 ICE Telephony calls in SIP Registrar mode

Configuration Needed on Cisco Unified Call Manager/SIP Registrar:

Important: Before configuring users and Directory numbers on ICE Server, the end users, 3rd party SIP devices with corresponding Directory Numbers need to be created first on CUCM/SIP Registrar.

Login to CUCM/SIP Registrar as admin to configure end users with SIP digest authentication and proceed to configure 3rd party SIP (advanced) devices associating end users and unique Directory Numbers.

Configuration needed on ICE Server:

1. Using ICE Desktop, login to Server as an administrator. In Settings->Call Manager, add your Call Manager/SIP Registrar to the ICE Server. Fill Mandatory fields of Name and IP Address along with port 5060. Description is an optional field
2. Configure users with assigned Directory Number(DN), Username, password. These fields need to match with what is configured on CUCM. Example : If DN 12345 has enduser *johndoe* with authpassword 12345 on CUCM, configure the same on ICE Server
3. Configure few channels with assigned DNs. DN, Username, and password need to match with what is configured on CUCM and need to be unique. **Note:** DNs/Username/password from one user cannot be repeated for other users as these DNs need to register to Call Manager.

Verify following ICE Telephony Call Flows after installation is complete:

1. Dial-in from a Cisco IP Phone(all supported models) calling assigned channel's number and confirm that ICE Android/ICE Desktop can hear audio when IP Phone presses *. IP Phone user can end PTT with #. IP Phone can hear audio when ICE Desktop/ICE Android speak on that assigned channel. IP Phone can end the call. Many IP Phones can dial to same channel but only one user will have floor control
2. Direct dial-in Call from IP Phones to ICE users using Directory number of ICE Android/Desktop users
3. Dial-out from ICE Desktop's/ICE Android's dial pad to to IP Phones; ICE users can also use Redial using call history tab
4. Private Calls between ICE (Desktop and Android) users using Telephone icon

7 ICE Telephony calls in SIP Trunk mode

7.1 Call Manager Configuration

Important: Before you proceed to configure users, channels and corresponding directory numbers on ICE Server, SIP Trunk needs to be configured on Call Manager first with two custom profiles: 'SIP Trunk Security' and 'SIP Trunk'.

1. Navigate to System->Security->SIP Trunk Security Profile and select the Standard Non Secured SIP Trunk Profile. Copy it and change Outgoing Transport Type as UDP. Save this as 'SIP Trunk Security Profile for ICE Telephony', and exit
2. Navigate to Device->Device Settings->SIP Profiles and select the Standard SIP Profile. Copy it and scroll down to Trunk Specific Configuration and set the Early Offer support for voice and

video calls Required Field to Best Effort no MTP needed (default is disabled). Enable SIP options flag (By default, it is not turned ON). Save this profile as 'SIP Trunk Profile to test ICE Telephony'

3. Within the SIP Profile page make sure that "**Session Refresh Method**" is set to Update instead of the default Invite.
4. Create a SIP Trunk pointing to ICE Telephony Server with port 5060 and make sure you select your newly created custom profiles in the trunk. Select Call Classification as OnNet, Save this SIP Trunk
5. Create the Route Pattern: Navigate to Call Routing-> Route/Hunt->Route Pattern. Create a route pattern with dial number pattern used in your organization. These numbers will be used for users and channels in ICE Server. For example, 2001 to 2999 Directory numbers configured need to be routed via SIP Trunk, configure the route pattern as 2XXX as shown below and save. You need to configure channel DNs and user DNs in this 2001 to 2999 range. Make sure this range and Route Pattern is unique and does not collide with DNs configured on IP Phones registered to the same Call Manager to avoid conflict.
6. Now Call Manager SIP Trunk is ready to route the calls to ICE Telephony Server which will route it to DNs configured on ICE Server

7.2 ICE Server Configuration

Important Step: If you have a Call Manager configured as SIP registrar, you need to delete the configuration before configuring it as SIP Trunk. Likewise, If you have a Call Manager configured for SIP Trunk , you need to delete the SIP Trunk configuration on Call Manager before toggling to re-use as SIP registrar as same ICE Telephony Server cannot be used in the SIP Trunk

1. Using Desktop 3.x Build, login to Server as Admin user. In Settings->Call Manager, add your Call Manager to the ICE Server as SIP Trunk. Fill Mandatory fields of Name and IP Address along with port 5060. Description is an optional field
2. Configure users with assigned Directory Number(DN) and selecting the Call Manager configured above
3. Configure few channels with assigned DNs and selecting the Call Manager field.

> **Notes:**
>
> - DNs from one user/channel cannot be re-used for other channels/users. Each DN needs to be unique.
> - SIP Trunk documentation on route pattern on Call Manager says * and # are used in special cases like below, so avoid * and # when configuring User and Channel DNs
> - The asterisk () *character can provide an extra digit for special dialed numbers. You can config-*

> *ure the route pattern* 411 to provide access to the internal operator for directory assistance.
> - The octothorpe (#) character generally identifies the end of the dialing sequence. The # character must be the last character in the pattern. The route pattern 9011819105555# routes or blocks an international number dialed from within the NANP. The # character after the last 5 identifies this as the last digit in the sequence.

8 Enable TLS/SRTP

This is the procedure for setting up Session Initiation Protocol (SIP) Transport Layer Security (TLS) and Secure Real-time Transport Protocol (SRTP) between a Cisco Unified Communications Manager (CUCM) and ICE Telephony Gateway.

Secure voice communication can be divided into two parts:

1. Secure signaling – ICE Telephony Gateway uses TLS to secure signaling over SIP
2. Secure Media – SRTP

8.1 Background Information

- TLS - TLS and its predecessor, Secure Sockets Layer (SSL), are cryptographic protocols that provide communication security over the Internet. TLS and SSL work on behalf of the underlying transport layer, whose segments carry encrypted data.
- Certificate Authority (CA) - Trusted entity that issues certificates: Cisco or a third-party entity.
- Device Authentication - Process that validates the identity of the device and ensures that the entity is what it claims to be before a connection is made.
- Encryption - Process of translating data into ciphertext that ensures the confidentiality of the information. Only the intended recipient can read the data. It requires an encryption algorithm and encryption key.
- Public/Private Keys - Keys that are used in encryption. Public keys are widely available, but private keys are held by their respective owners. Asymmetrical encryption combines both types.

8.2 Generate a self-signed certificate file (.pem)

Either a 3rd party certificate, generated by a certificate authority, or a self-signed certificate is required to establish a TLS connection between CUCM and the ICE Telephony Gateway. In the following example a self-signed certificate is generated using the OpenSSL command line tool.

1. Open the OpenSSL command line tool.
2. Enter the following command:

```
openssl req -x509 -newkey rsa:4096 -keyout icegwkey.pem -out icegw.pem
    -days 365 -nodes
```

3. From the resulting output, enter the required certificate information, see the example below.

```
kammoham@MacBook-Pro ~ % openssl req -x509 -newkey rsa:4096 -keyout icetkey.pem -out icet.pem -days 365 -nodes
Generating a 4096 bit RSA private key
.................................++
.......................................++
writing new private key to 'icetkey.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) []:US
State or Province Name (full name) []:IL
Locality Name (eg, city) []:Chicago
Organization Name (eg, company) []:DKG
Organizational Unit Name (eg, section) []:ICE
Common Name (eg, fully qualified host name) []:192.168.0.65
Email Address []:icet@dkg.com
kammoham@MacBook-Pro ~ %
```

   - Country Name
   - State or Province Name
   - Locality Name
   - Organization Name
   - Common Name = Enter the IP address of the ICE Telephony Gateway. That will be the destination address for the SIP Trunk created later in this process.
   - Email Address

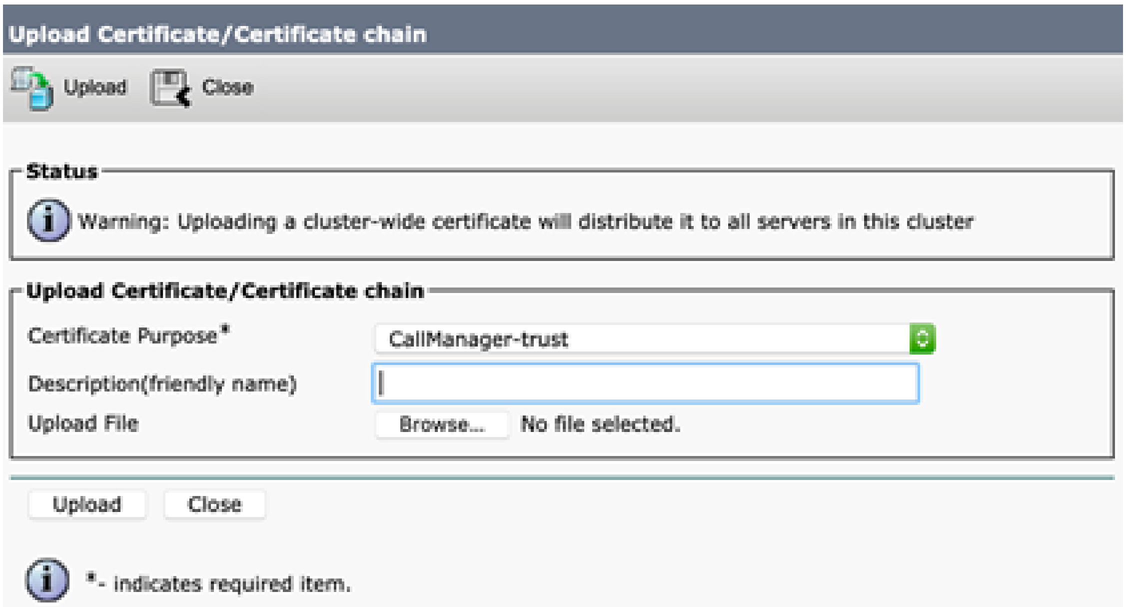
4. Enter the following command: `openssl`
5. Two PEM files are generated:

   - `icegwkey.pem`
   - `icegw.pem`

8.3 Upload the certificate file to CUCM

1. Log in to the CUCM 'Cisco Unified OS Administration' page.
2. Navigate to Security > Certificate Management > Find, and click 'Upload Certificate/Certificate chain':

3. From the 'Upload Certificate/Certificate chain' screen:

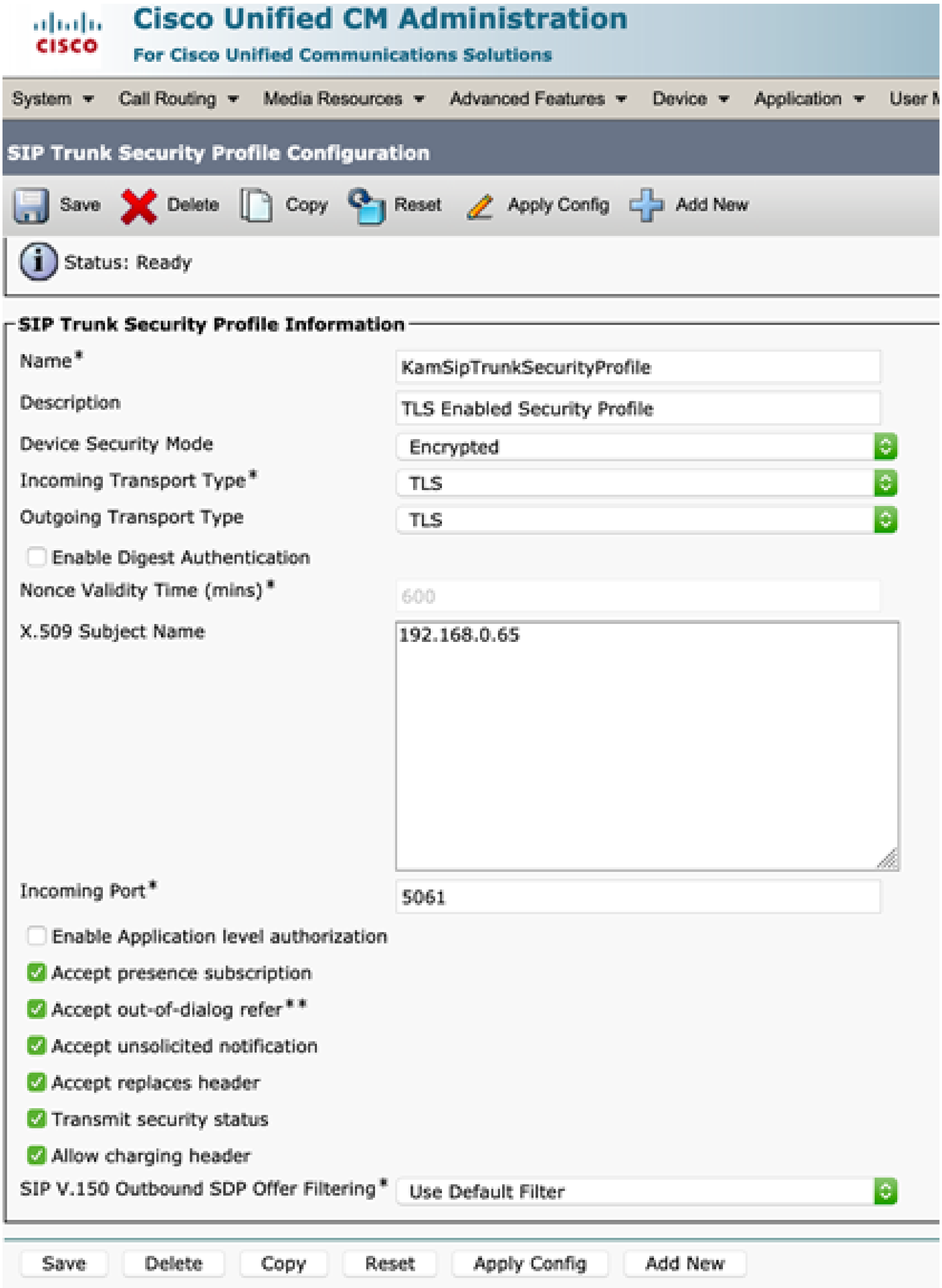


- Certificate Purpose = CallManager-trust
- Upload File = Click 'Browse', then select the .pem certificate file generated prior, in this example it is the 'icet.pem' file from the section above.

4. Click 'Upload'.
5. Click 'Close'.

8.4 Create a SIP Trunk Security Profile

1. From the 'Cisco Unified CM Administration' page, navigate to System > Security > SIP Trunk Security Profile > Add New.
2. Select 'Add New':



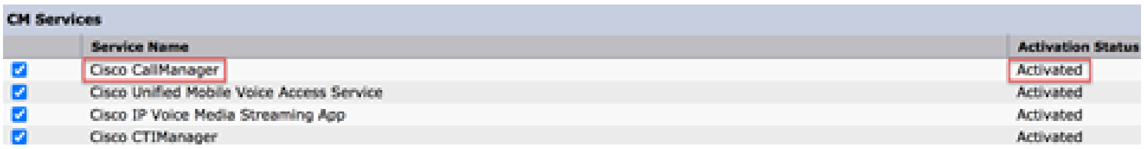
- X.509 Subject Name = Enter the IP address of the ICE Telephony Gateway. Must be the same IP address entered for ‘Common Name’ in the *Generate a self-signed certificate fiel (.pem)* section above.

3. Click ‘Apply Config’.
4. Click ‘Save’.

8.5 Create a SIP Trunk

8.5.1 Verify the Cisco CallManager Service is activated on CUCM

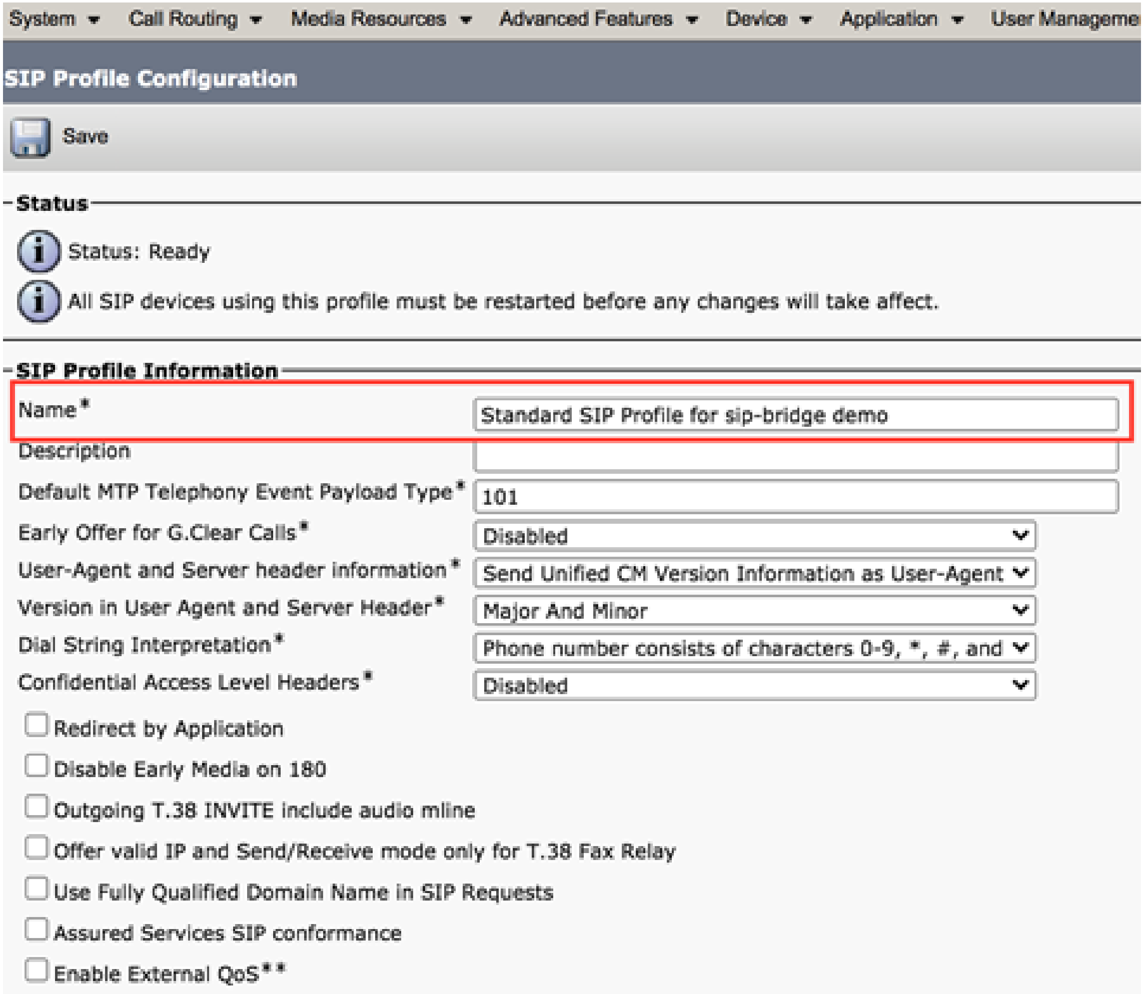
1. From the ‘Cisco Unified CM Administration’ page, go to the ‘Navigation’ field at the top, right corner and select ‘Cisco Unified Serviceability’.
2. Click ‘Go’.
3. Under ‘Tools’, click ‘Service Activation’.
4. Verify that ‘Cisco CallManager’ is activated:


CM Services

	Service Name	Activation Status
☑	Cisco CallManager	Activated
☑	Cisco Unified Mobile Voice Access Service	Activated
☑	Cisco IP Voice Media Streaming App	Activated
☑	Cisco CTIManager	Activated

8.5.2 Create a SIP Profile

1. From the ‘Cisco Unified CM Administration’ page, navigate to Device > Device Settings > SIP Profile.
2. Click ‘Standard SIP Profile’.
3. From the ‘SIP Profile Configuration’ screen, click ‘Copy’.
4. Complete the rest of the SIP Profile as needed:

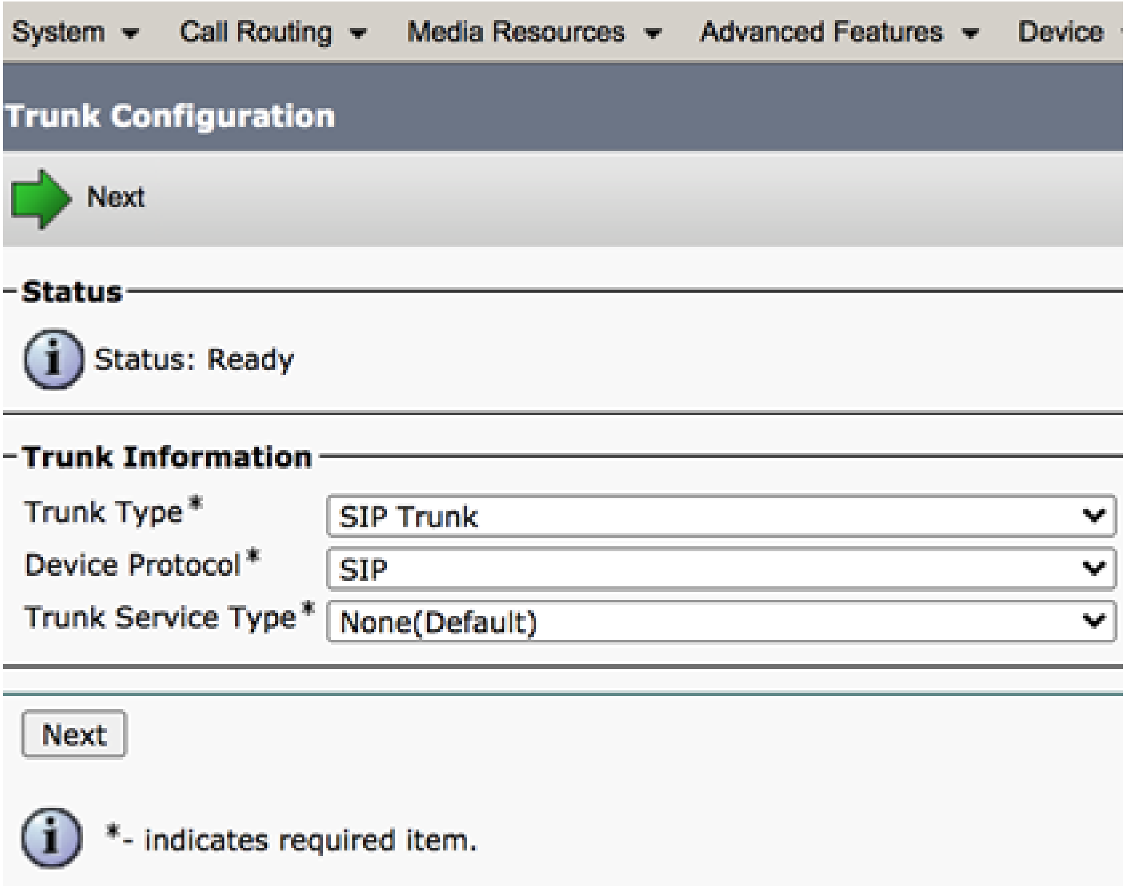


- Select 'Allow Presentation Sharing using BFCP', if BFCP (Dual video / presentation sharing) is required.
- Select 'Use Fully Qualified Domain in SIP Requests', if needed.

5. Click 'Save'.

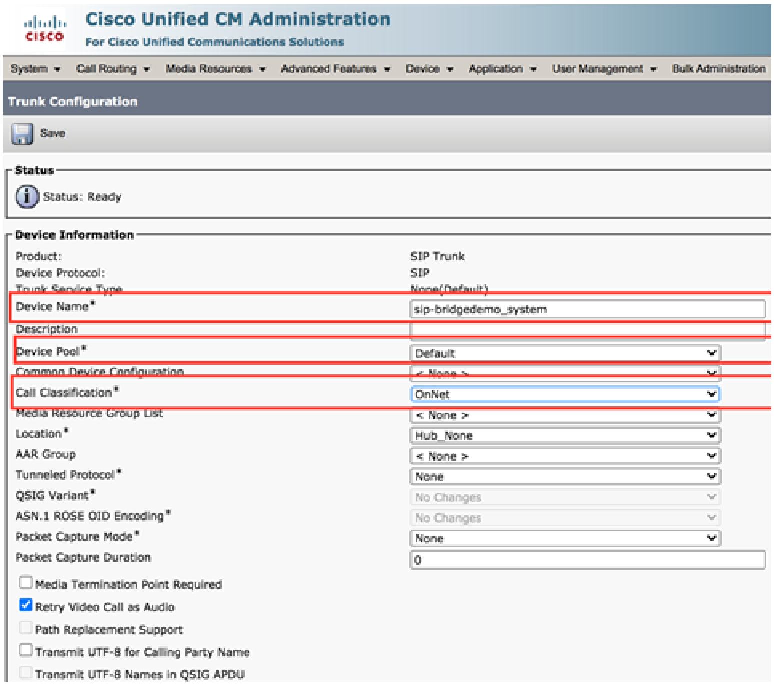
8.5.3 Create a SIP Trunk Device

1. From the 'Cisco Unified CM Administration' page, navigate to Device > Trunk.
2. Click 'Add New'.

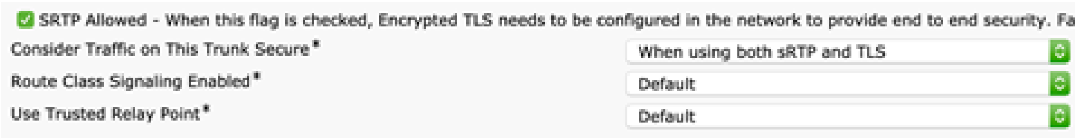


- Trunk Type = SIP Trunk

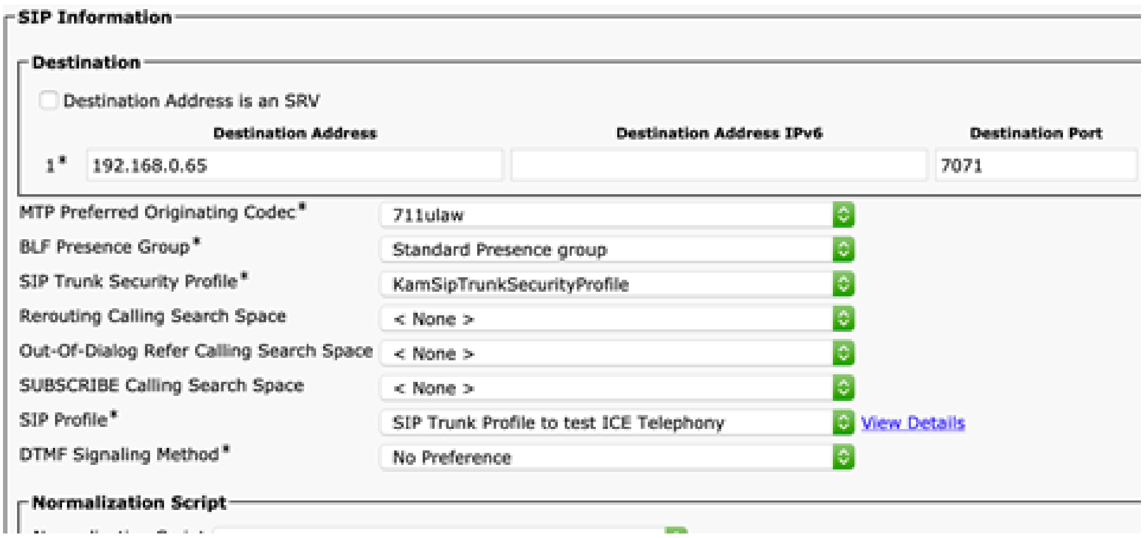
3. Click ‘Next’.
4. Configure the ‘Device Information’ fields as needed:



- SRTP Allowed = Enable as shown below:



- SIP Information = Enable options as described below:



- Destination Address = Enter the IP address of the ICE Telephony Gateway. Must be the same IP address entered for ‘Common Name’ in the *Generate a self-signed certificate fiel*

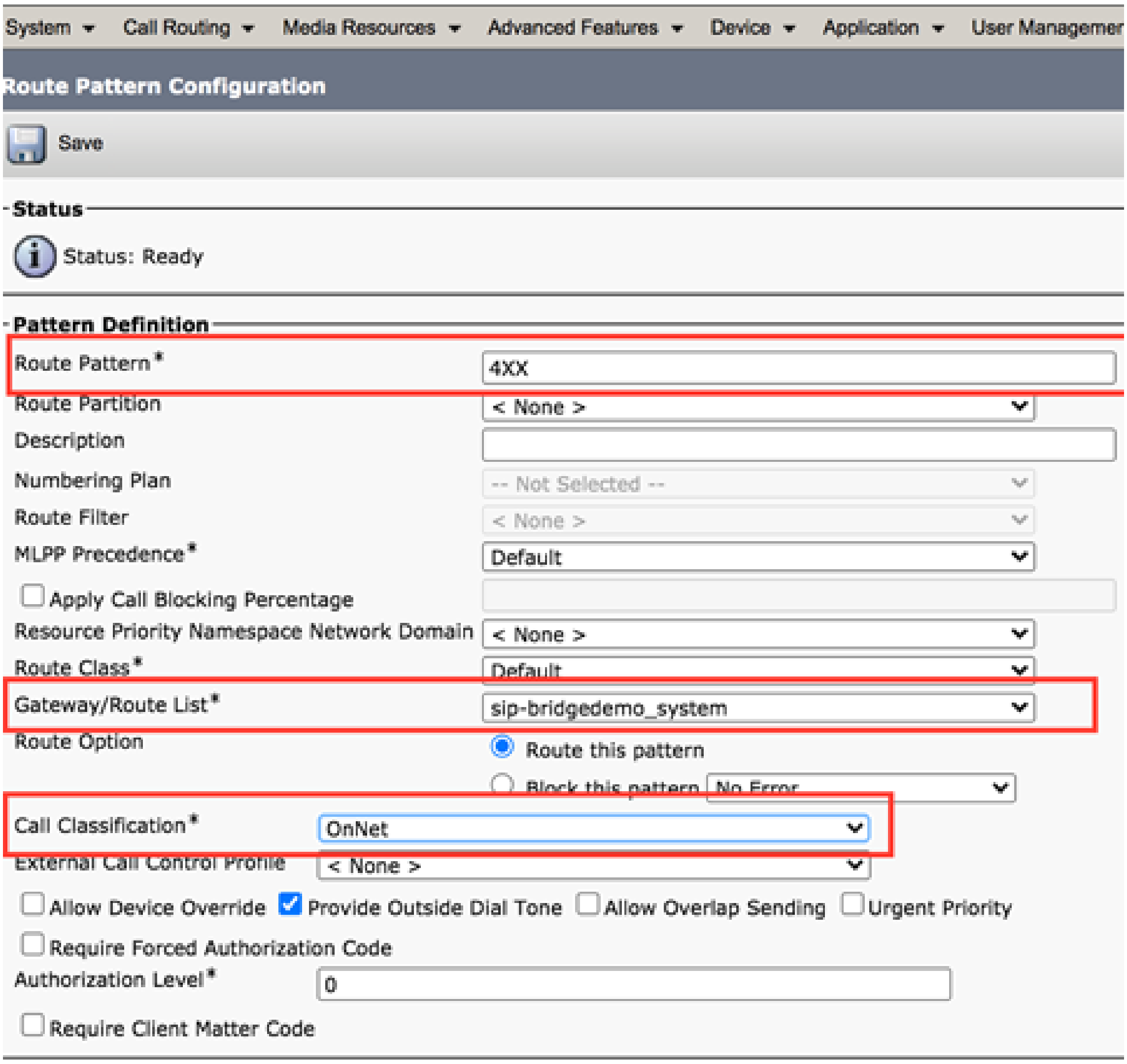
(.pem) and the *Create a SIP Trunk Security Profile* sections above.

- Destination Port = 5061. Must match the `sip_tls_port` property configured at `$ICET_HOME/conf/icet_conf.json`. This is the SIP TLS port on which the ICE Telephony Gateway is listening for incoming TLS connections.
- SIP Trunk Security Profile = Enter the name of the SIP Trunk that was created above.

5. Click 'Save'.
6. Click 'Reset'.

8.5.4 Create a Route Pattern for the SIP Trunk

1. From the 'Cisco Unified CM Administration' page, navigate to Call Routing > Route/Hunt > Route Pattern.
2. Click 'Add New'.
3. Configure Route Pattern as follows:

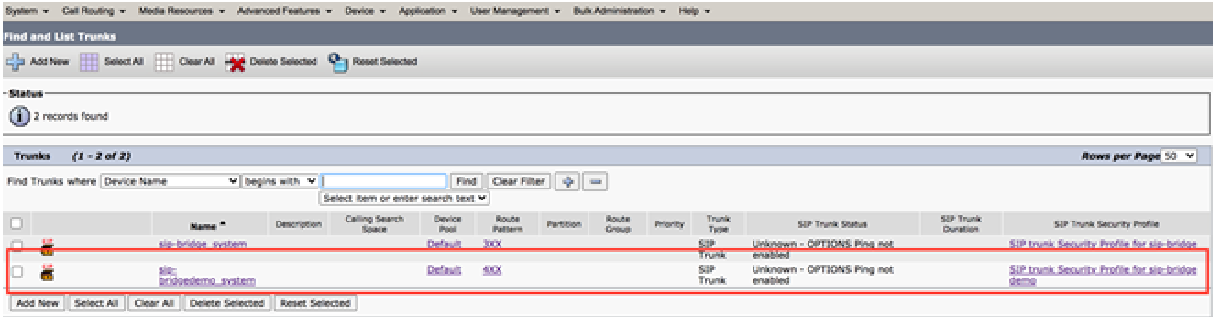


- Route Pattern = 4XX, which means all 3-digit calls starting with 4 will be sent to sip-bridge via this SIP Trunk.
- Multiple Route Patterns can be configured for a SIP Trunk.

4. Click 'Save'.

8.5.5 Check the SIP Trunk status

1. From the 'Cisco Unified CM Administration' page, navigate to Device > Trunk.
2. Use the search filter to verify the SIP Trunk exists.



8.6 Enable CUCM to operate in 'mixed-mode'

In order to enable CUCM to accept calls both in secure and insecure mode, the admin needs to turn on the 'mixed-mode' flag in the CUCM via the command line interface.

1. Use Secure Shell (SSH) protocol to get in to the CUCM command line shell and apply the following command:

```
admin:utils ctl set-cluster mixed-mode
```

2. CUCM will prompt the user to confirm this operation:

```
This operation will set the cluster to Mixed mode. Do you want to
    continue? (y/n):
```

3. Press 'y' to continue and the user should see the following response:

```
Moving Cluster to Mixed Mode
Cluster set to Mixed Mode
Please Restart Cisco Tftp, Cisco CallManager and Cisco CTIManager
    services on all nodes in the cluster that run these services.
```

8.7 Configure ICE Telephony Gateway to support TLS/SRTP

The `preferred_signalling_protocol` property in `$ICET_HOME/conf/icet_conf.json` need to be set to `tls` to make/receive secure calls to/from CUCM.

The ICE Telephony Gateway default TLS port is 5061 in `icet_conf.json`.

The following changes are needed to be applied in the `$ICET_HOME/conf/pjsip_acfg.json` file:

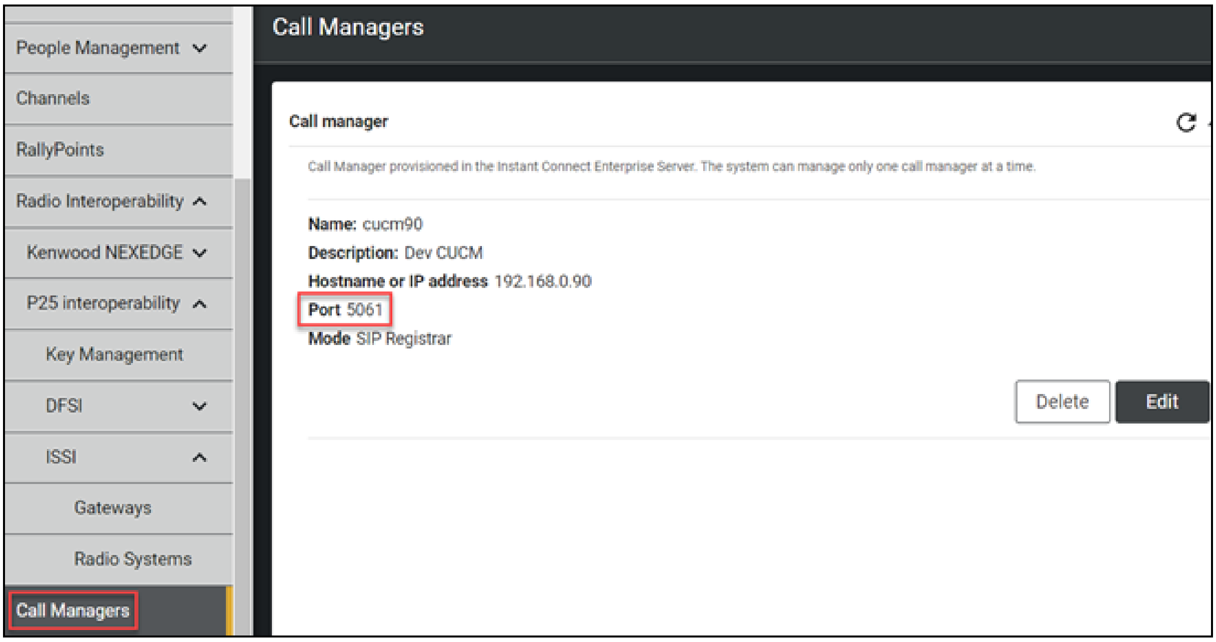
```
"srtpUse":             1,
"srtpSecureSignaling": 1,
"TlsConfig":           {
```

```
    "CaListFile":           "",
    "certFile":             "$ICET_HOME/etc/icet.pem",
    "privKeyFile":          "$ICET_HOME/etc/icetkey.pem",
    "password":             "",
    "CaBuf":                "",
    "certBuf":              "",
    "privKeyBuf":           "",
    "method":               33,
    "ciphers":              [ ],
    "verifyServer":         false,
    "verifyClient":         false,
    "requireClientCert":    false,
    "msecTimeout":          0,
    "qosType":              3,
    "qosParams":            {
       "qos.flags":            1,
       "qos.dscp_val":         24,
       "qos.so_prio":          0,
       "qos.wmm_prio":         0
    },
    "qosIgnoreError":       true
}
```

> Note: As shown above, the values for the `certFile` and `privKeyFile` attributes are the files created in the *Generate a self-signed certificate file* section above using the OpenSL command tool. If a password was used while creating the certificate (private key) file, then that password should be set here in the `password` attribute.

8.8 ICE Desktop

When creating a new Call Manager in the ICE Desktop, the port number should be whatever port number assigned in the CUCM SIP Trunk to handle TLS calls, by default this port number is 5061.

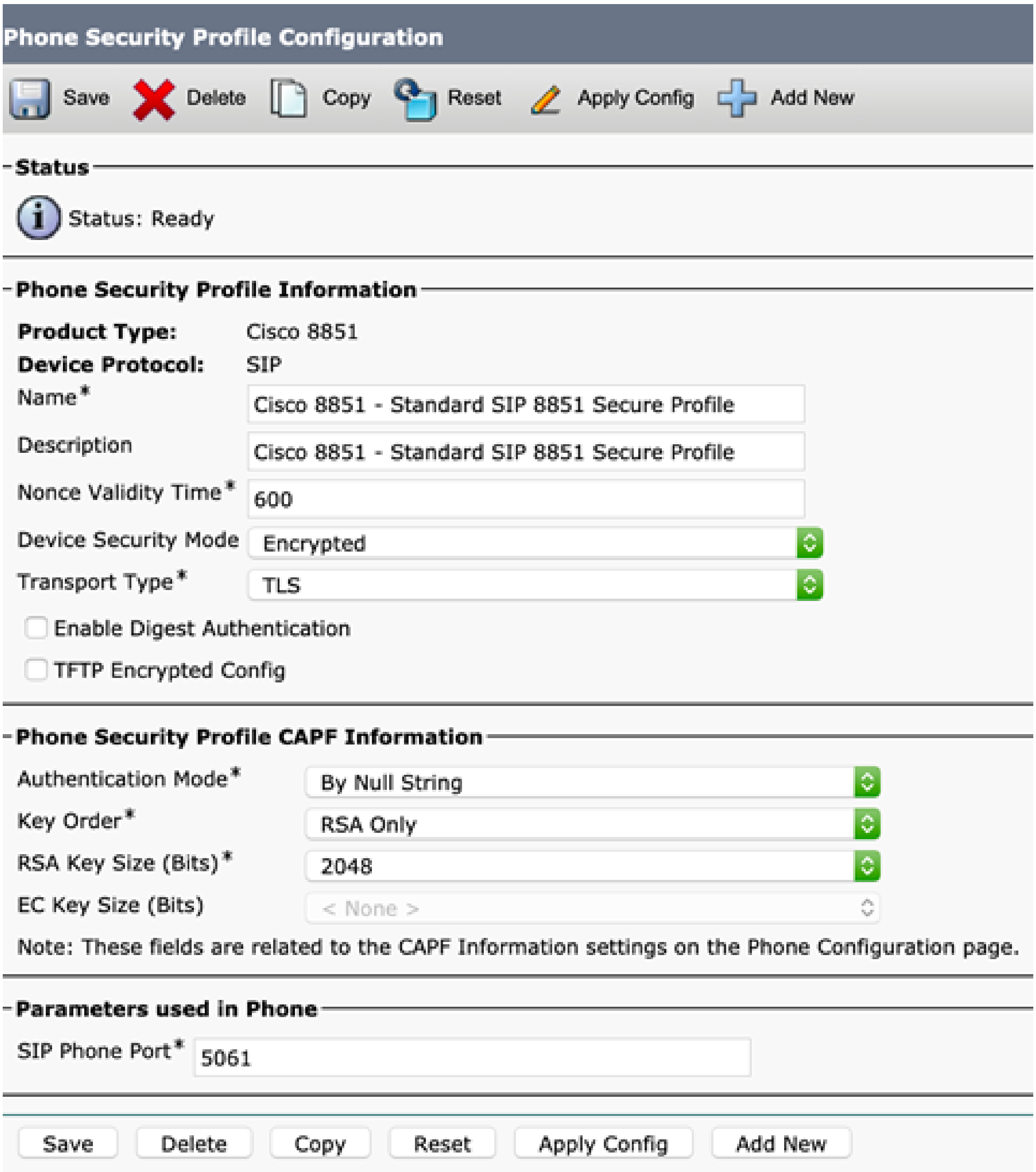


8.9 Establish secure communication between a Cisco IP Phone and CUCM

> Note: For this process, the Cisco IP Phone model 8851 is used as an example. The process may vary if using other phone models.

8.9.1 Create a Phone Security Profile

1. From the 'Cisco Unified CM Administration' page, navigate to System > Security > Phone Security Profile.
2. Click 'Add New'.
3. Select the appropriate IP Phone model from the 'Phone Security Profile Type' drop-down menu:

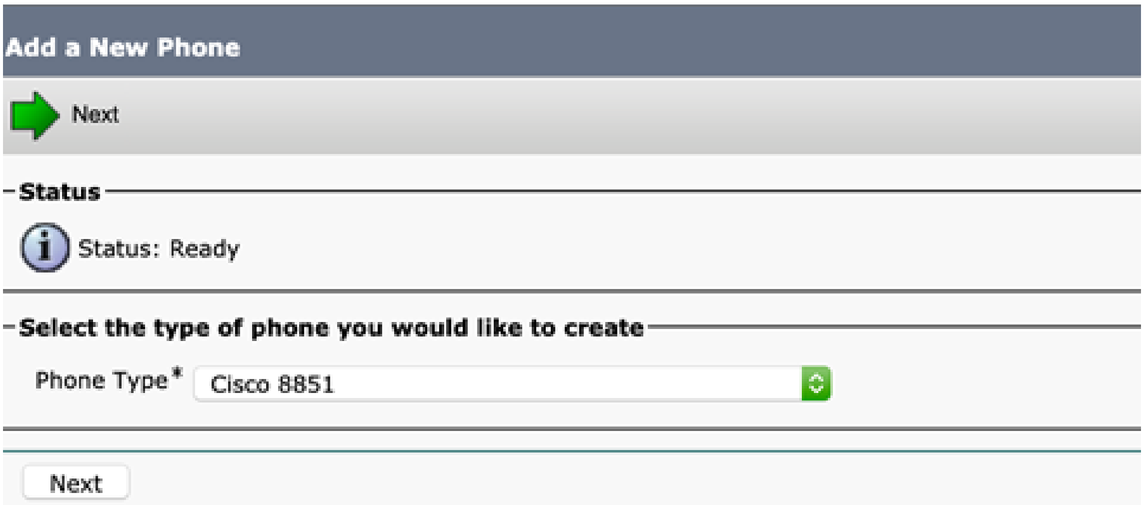


4. Click ‘Save’.
5. Click ‘Apply Config’.
6. Click ‘Reset’.

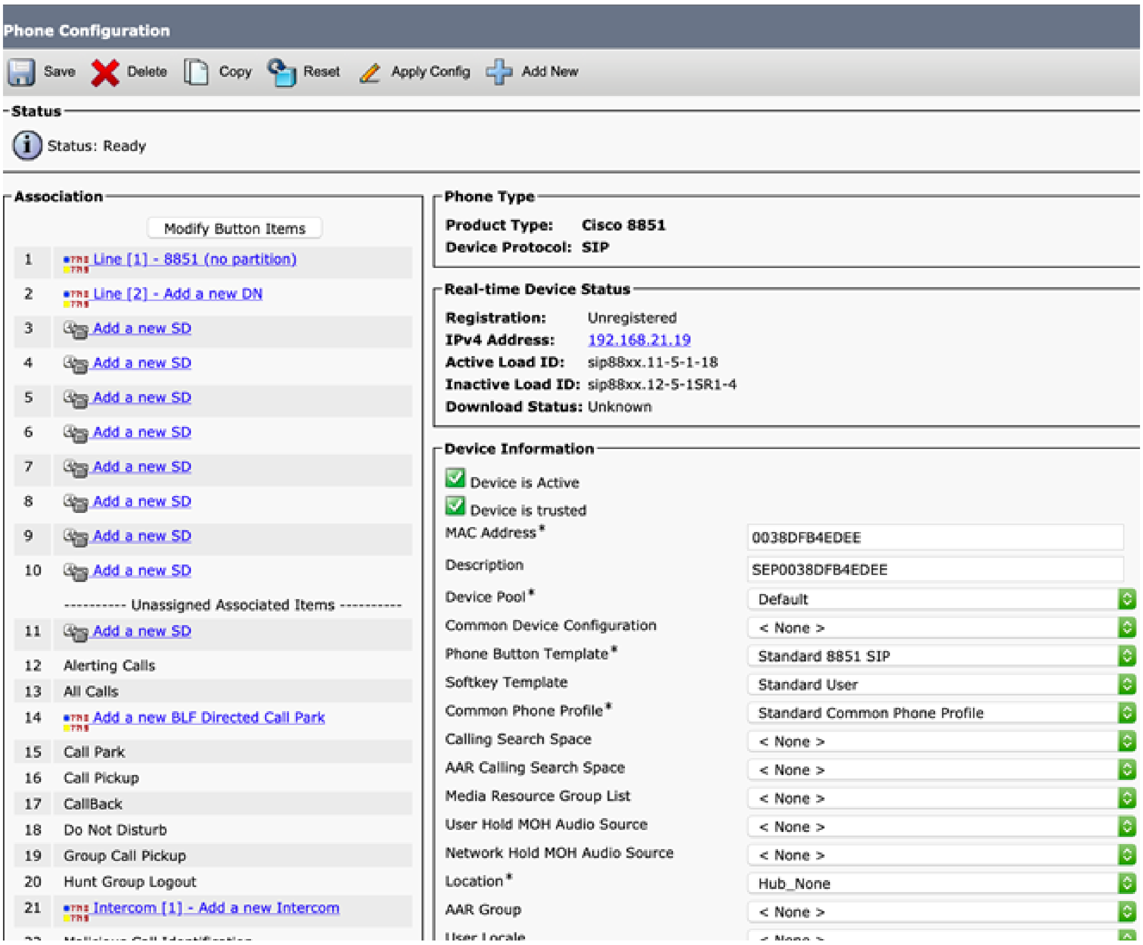
8.9.2 Create a Phone Device using the Phone Security Profile

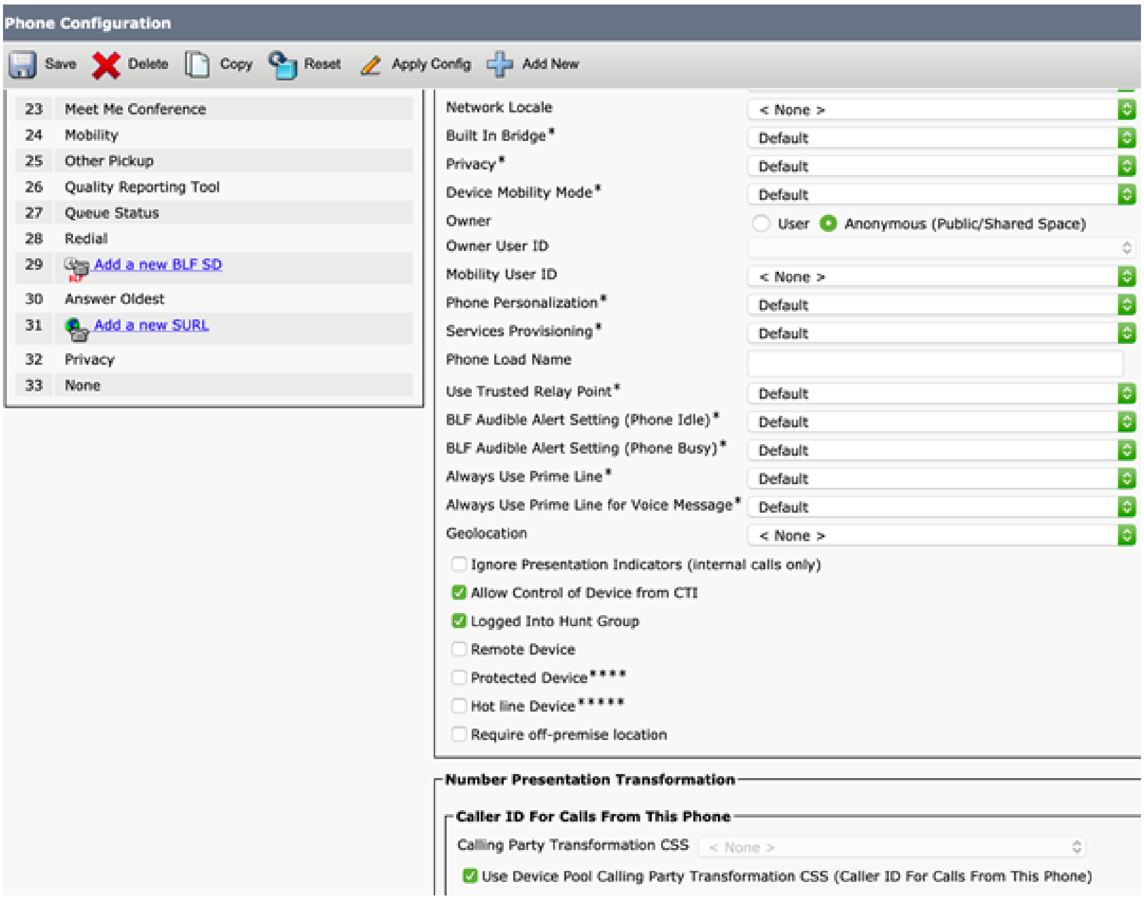
1. From the ‘Cisco Unified CM Administration’ page, navigate to Device > Phone.

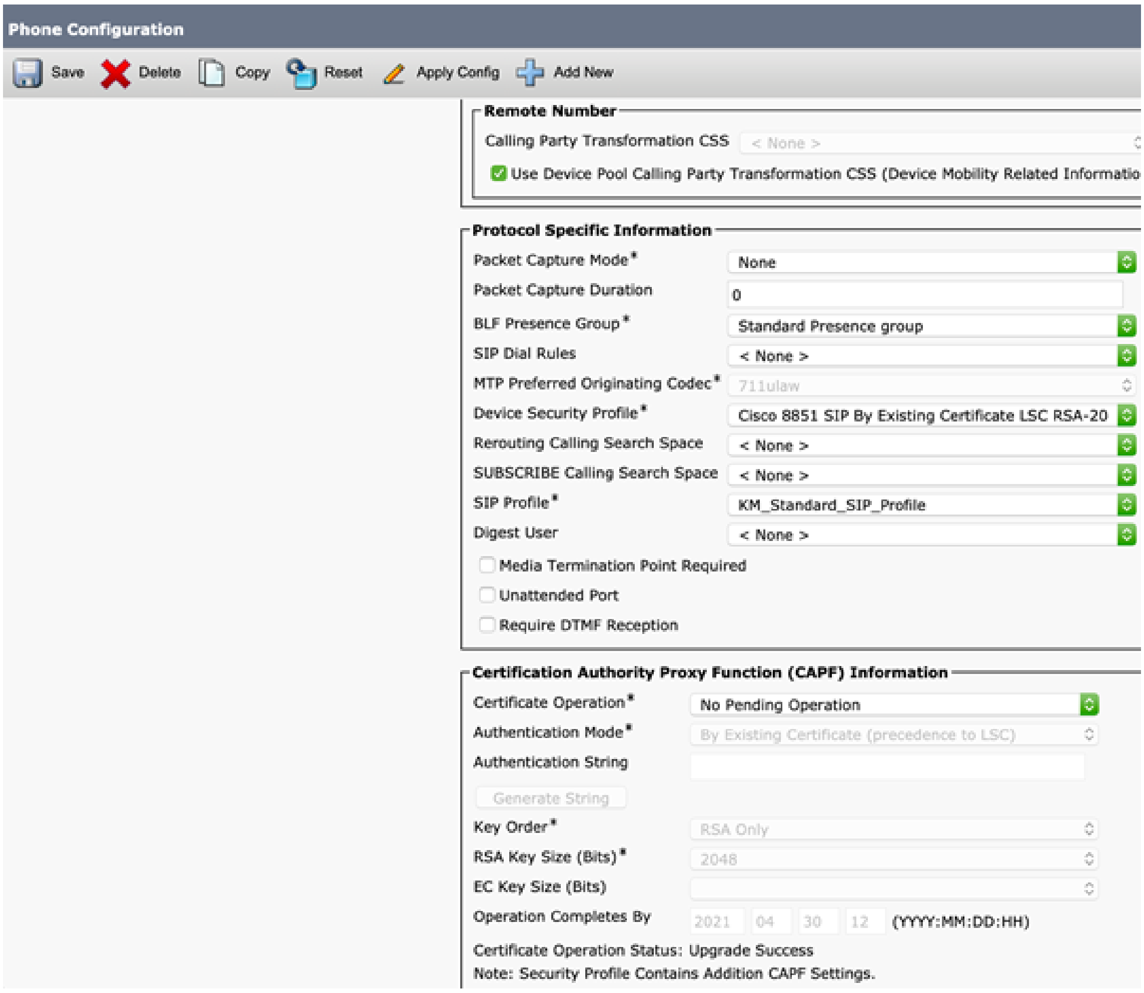
2. Click ‘Add New’.
3. Select the appropriate Phone Type:

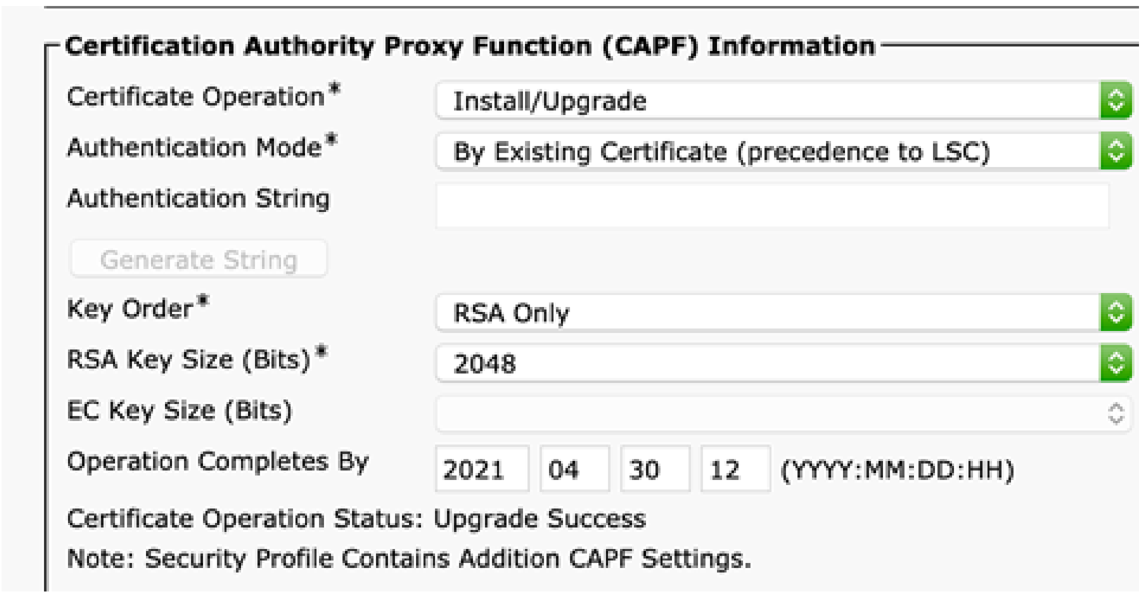


4. Click ‘Next’ and use the following images as a guide:




Phone Configuration
Save
Delete
Copy
Reset
Apply Config
Add New
23 Meet Me Conference
24 Mobility
25 Other Pickup
26 Quality Reporting Tool
27 Queue Status
28 Redial
29 Add a new BLF SD
30 Answer Oldest
31 Add a new SURL
32 Privacy
33 None
Network Locale < None >
Built In Bridge* Default
Privacy* Default
Device Mobility Mode* Default
Owner User Anonymous (Public/Shared Space)
Owner User ID
Mobility User ID < None >
Phone Personalization* Default
Services Provisioning* Default
Phone Load Name
Use Trusted Relay Point* Default
BLF Audible Alert Setting (Phone Idle)* Default
BLF Audible Alert Setting (Phone Busy)* Default
Always Use Prime Line* Default
Always Use Prime Line for Voice Message* Default
Geolocation < None >
Ignore Presentation Indicators (internal calls only)
Allow Control of Device from CTI
Logged Into Hunt Group
Remote Device
Protected Device****
Hot line Device*****
Require off-premise location
Number Presentation Transformation
Caller ID For Calls From This Phone
Calling Party Transformation CSS < None >
Use Device Pool Calling Party Transformation CSS (Caller ID For Calls From This Phone)





5. Leave the remaining fields as default.

6. Click 'Save'.
7. Click 'Apply Config'.
8. Click 'Reset'.

8.10 Configure the Docker container 'env' file

The Docker container 'env' file requires some additional configuration:

- To enable TLS:

```
ICET_CONF__tls_supported=true
ICET_CONF__preferred_signalling_protocol"="tls
```

- To ensure `pjsip_acfg.json` is mounted from the home folder, along with the `icet.pem` and `icetkey.pem` certificate files:

```
docker run
--detach \
--net=host \
--name telephony \
--env-file env \
--volume ${GATEWAY_TLS_CERT_FILE_PATH}:${GATEWAY_TLS_CERT_FILE_PATH} \
--volume ${GATEWAY_TLS_PRIVATE_KEY_FILE_PATH}:${
    GATEWAY_TLS_PRIVATE_KEY_FILE_PATH} \
--volume /home/iceadmin/telephony-logs:/home/telephony/icet/logs \
--volume /home/iceadmin/telephony-cores:/tmp \
--restart always \
--ulimit core=-1 \
--log-driver json-file \
--log-opt max-size=1g instantconnect/ice/gateway:3.5.7732
```

- Then restart the container, see the *Running the container* section above for the restart command.