Configuration Guide for Google CCAI Call Recording Using Avaya Session Border Controller 10.2.1.1-104-25336

TekVizion

Table of Contents

1 Audience

This document is intended for the SIP Trunk customer's technical staff and Value-Added Reseller (VAR) having installation and operational responsibilities.

1.1 Introduction

This configuration guide describes configuration steps for **Google CCAI Call Recording** using **Avaya Session Border Controller v10.2.1.1-104-25336**.

1.1.1 TekVizion Labs

TekVizion Labs™ is an independent testing and verification facility offered by TekVizion, Inc. TekVizion Labs offers several types of testing services including:

- Remote Testing – provides secure, remote access to certain products in TekVizion Labs for pre-Verification and ad hoc testing.
- Verification Testing – Verification of interoperability performed on-site at TekVizion Labs between two products or in a multi-vendor configuration.
- Product Assessment – independent assessment and verification of product functionality, interface usability, assessment of differentiating features as well as suggestions for added functionality, stress, and performance testing, etc.

TekVizion is a systems integrator specifically dedicated to the telecommunications industry. Our core services include consulting/solution design, interoperability/Verification testing, integration, custom software development and solution support services. Our services help service providers achieve a smooth transition to packet-voice networks, speeding delivery of integrated services. While we have expertise covering a wide range of technologies, we have extensive experience surrounding our practice areas which include SIP Trunking, Packet Voice, Service Delivery, and Integrated Services.

The TekVizion team brings together experience from the leading service providers and vendors in telecom. Our unique expertise includes legacy switching services and platforms, and unparalleled product knowledge, interoperability, and integration experience on a vast array of VoIP and other next-generation products. We rely on this combined experience to do what we do best: help our clients advance the rollout of services that excite customers and result in new revenues for the bottom line. TekVizion leverages this real-world, multi-vendor integration and test experience and proven processes to offer services to vendors, network operators, enhanced service providers, large enterprises and other professional services firms. TekVizion's headquarters, along with a state-of-the-art test lab and Executive Briefing Center, is located in Plano, Texas.

For more information on TekVizion and its practice areas, please visit TekVizion Labs website.

2 SIP Trunking Network Components

The network for the SIP trunk reference configuration is illustrated below and is representative of Google CCAI Call Recording with Avaya Session Border Controller (ASBC) v v10.2.1.1-104-25336configuration.

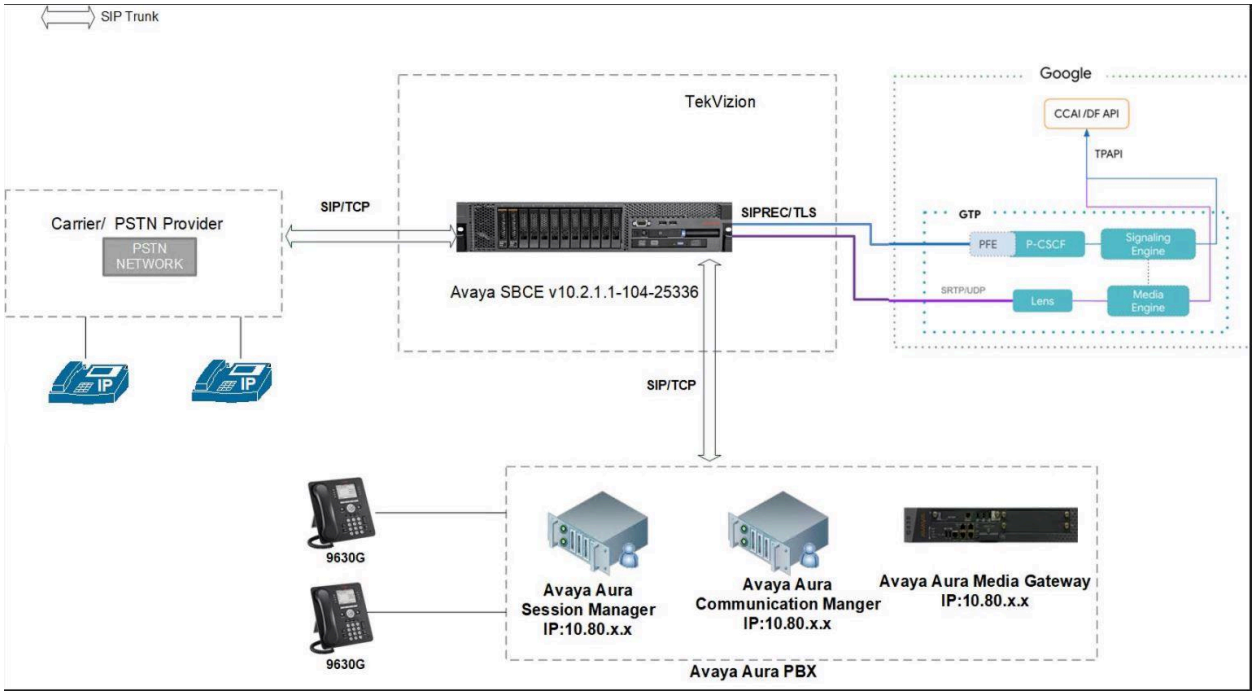

Figure 1: SIP Trunk Lab Reference Network

The lab network consists of the following components:

- Google CCAI Cloud Environment
- Avaya Session Border Controller (ASBC) v10.2.1.1-104-25336
- OnPrem PBX (Avaya Aura PBX).

3 Hardware Components

- Running on ESXi- 7.0.3: Avaya SBC v10.2.1.1-104-25336

4 Software Requirements

- Avaya SBC v10.2.1.1-104-25336
- OnPrem PBX (Avaya Aura PBX)

5 Google CCAI Certified Avaya SBC Version

Table 1 – Google CCAI Certified Avaya Version

Google CCAI Certified Avaya Version	
Avaya SBC	8.1.3.2-38-22279
Avaya SBC	10.2.0.0-86-24077
Avaya SBC	10.2.1.1-104-25336

6 Features

6.1 Features Tested for Google CCAI Call Recording

- Basic Inbound calls
- Call Hold and Resume
- Call Transfer
- Conference

6.2 Features Not Tested for Google CCAI Call Recording

- None

6.3 Caveats and Limitations

DTLS	DTLS towards Google CCAI is not supported
Blind Transfer	Avaya PBX does not support blind transfer. This test case is performed by ringing transfer
Long duration call	Avaya SBC does not send session refresh RE-INVITE. Google CCAI sends session refresh every 60 minutes using RE-INVITE

7 Configuration

7.1 Configuration Checklist

Below are the steps that are required to configure Avaya SBC.

Table 2 – Avaya SBC Configuration Steps

Step	Description	Reference
Step 1	Avaya SBC Login	Section 7.4.1
Step 2	Server Interworking	Section 7.4.2
Step 3	SIP Servers	Section 7.4.3
Step 4	Topology Hiding	Section 7.4.4
Step 5	Routing	Section 7.4.5
Step 6	Recording Profile	Section 7.4.6
Step 7	Session Policies	Section 7.4.7
Step 8	Session Flows	Section 7.4.8
Step 9	Signaling Manipulation	Section 7.4.9
Step 10	Signaling Rules	Section 7.4.10
Step 11	End Point Policy Groups	Section 7.4.11
Step 12	Media Interface	Section 7.4.12
Step 13	Network Management	Section 7.4.13
Step 14	Signaling Interface	Section 7.4.14
Step 15	End Point Flow	Section 7.4.15
Step 16	TLS Configuration	Section 7.4.16

7.2 IP Address Worksheet

The specific values listed in the table below and in subsequent sections are used in the lab configuration described in this document are for **illustrative purposes only**.

Table 3 – IP Address Worksheet

Component	IP Address
Google CCAI	
Signaling	us.telephony.goog
Media	74.125.X.X
OnPrem PBX	
LAN IP Address	10.70.X.X
Avaya SBC	
LAN IP Address	10.64.X.X
WAN IP Address	192.65.X.X

7.3 Google CCAI API Configuration

Below link can be referred to configure Google CCAI API configuration for Call recording.

---------Link to be provided by Google team----------

7.4 Avaya ASBC Configuration

The following is the example configuration of Avaya SBC for Google CCAI Call Recording.

7.4.1 Avaya SBC Login

- Log into Avaya Session Border Controller (ASBC) web interface by typing **"https://X.X.X.X/sbc".**
- Enter the **Username** and **Password**
- Click **Log In**

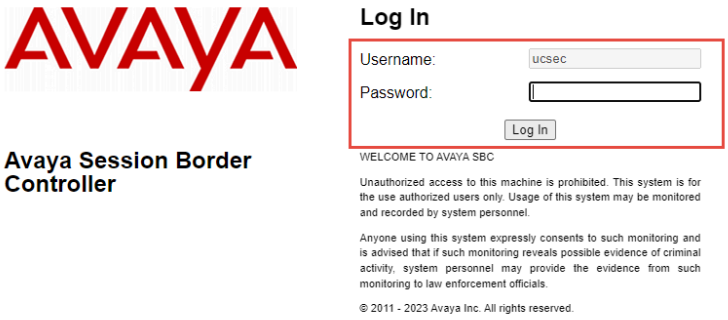

Figure 2: Avaya SBC Login

- Device, select **Name (SA)** from drop down to expand the configuration for Avaya SBC.

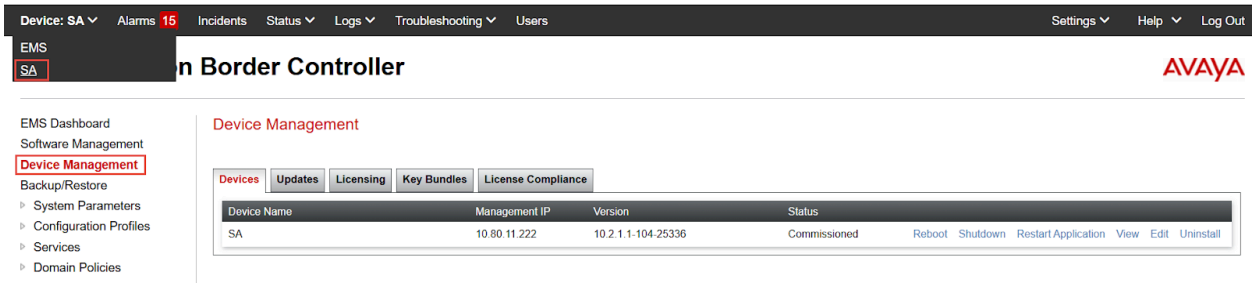

Figure 3: Selection of Avaya SBC Device

7.4.2 Server Interworking

Server Interworking for **Avaya Aura Session Manager (SM)**

- Navigate: **Configuration Profiles > Server Interworking**
- Select the default Interworking Profile avaya-ru, click Clone
- Set Clone Name: **AvayaSM10.2**
- Click **Finish**

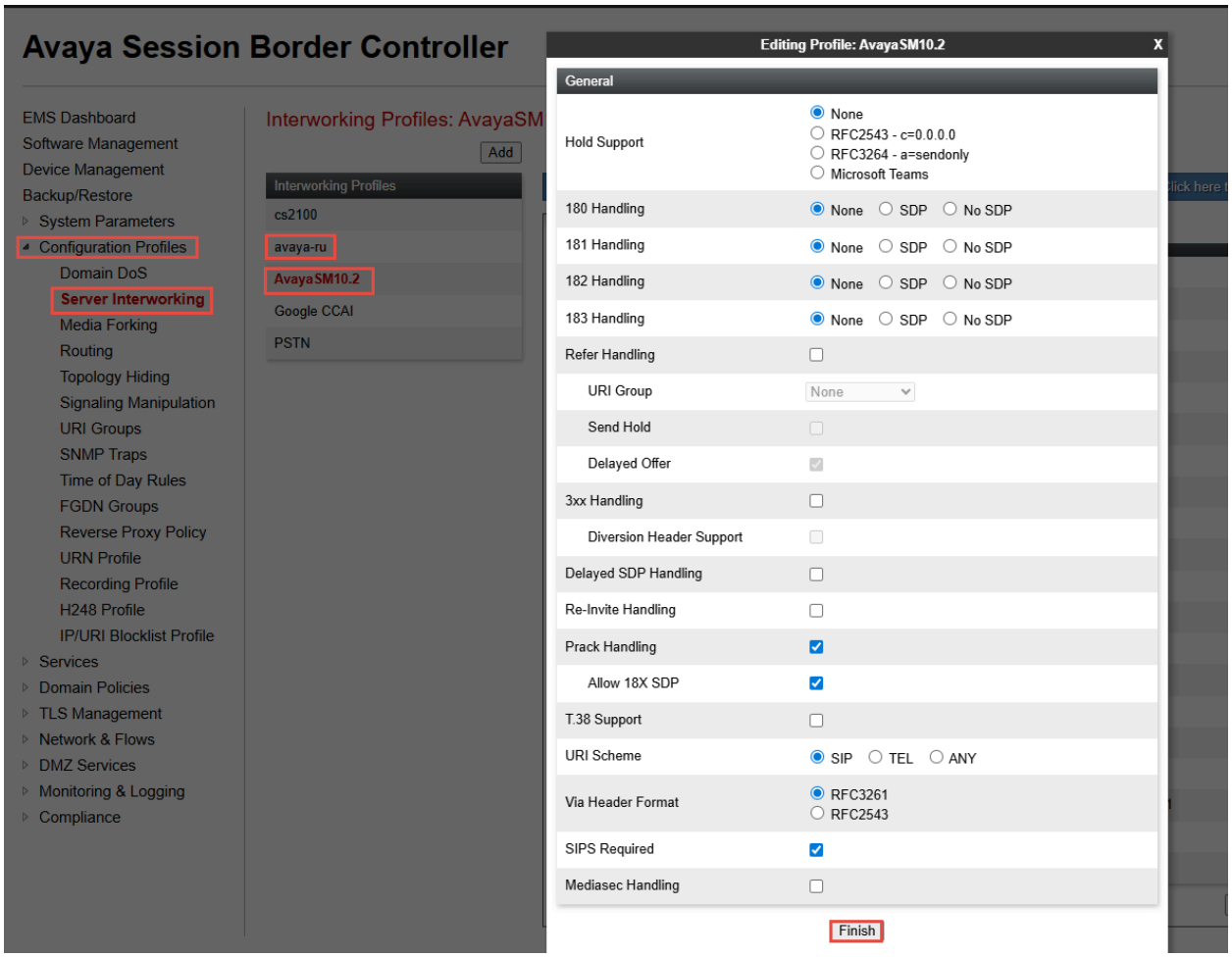

Figure 4: Server Interworking Profile for Avaya Aura SM

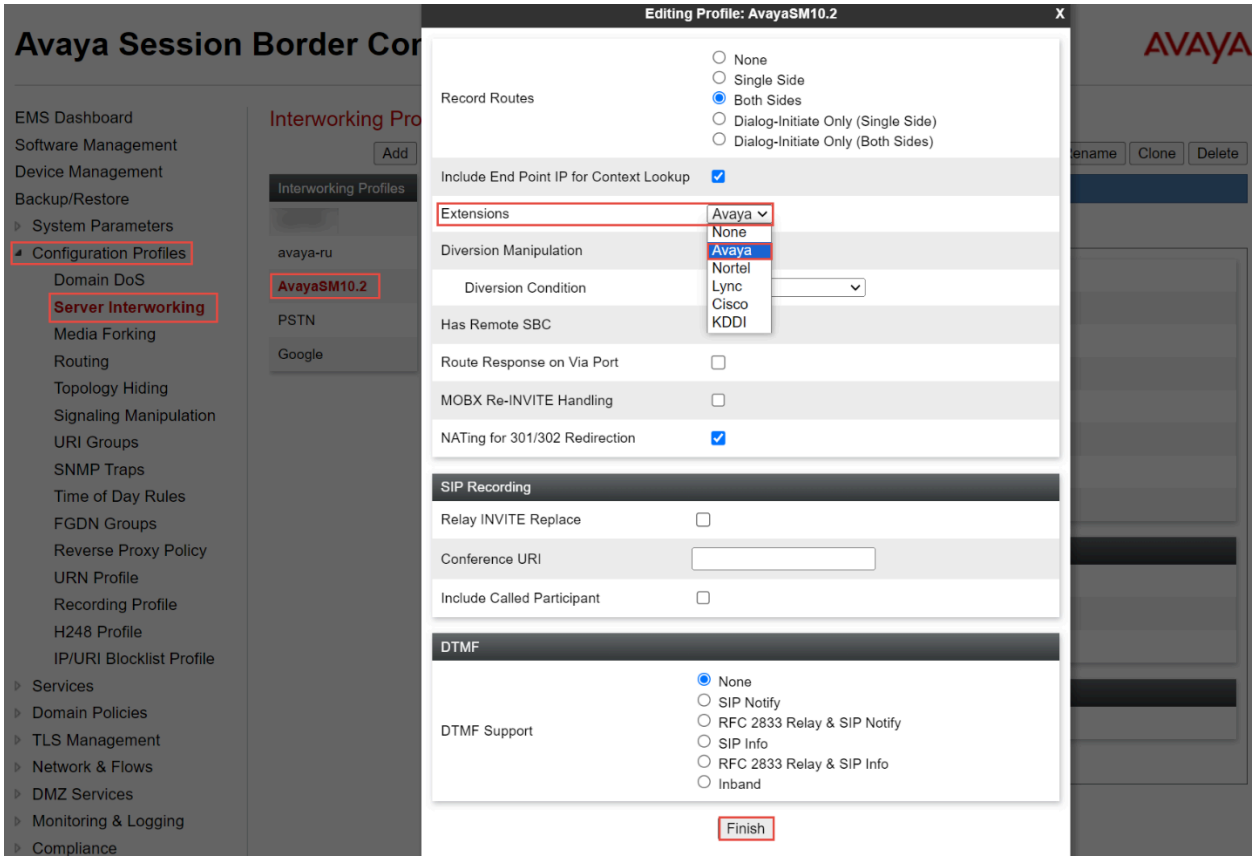

Figure 5: Server Interworking Profile for Avaya Aura SM (Cont.)

Server Interworking for **Google CCAI**

- Repeat the same procedure to create the Interworking Profile towards **Google CCAI.**

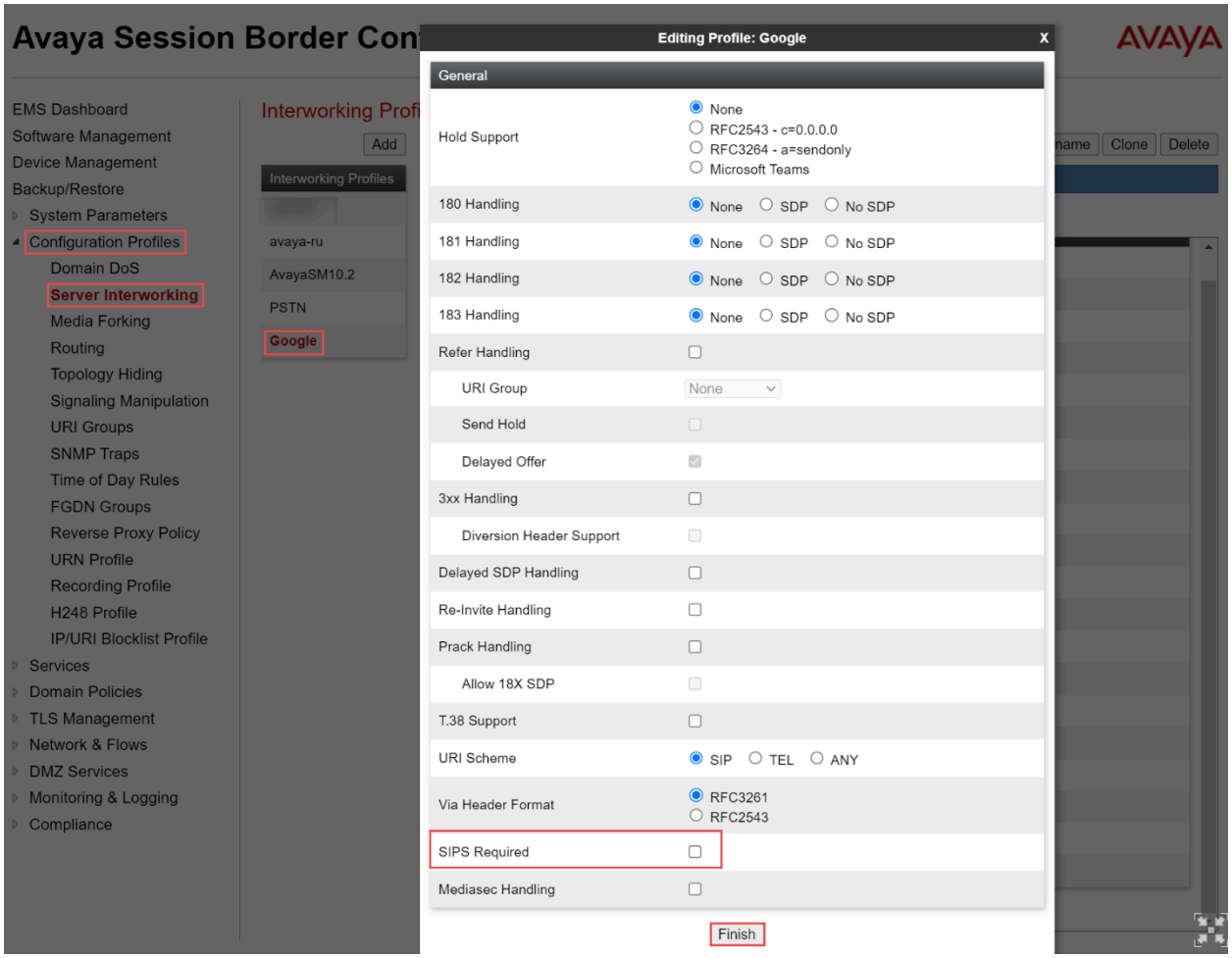

Figure 6: Server Interworking Profile for Google CCAI

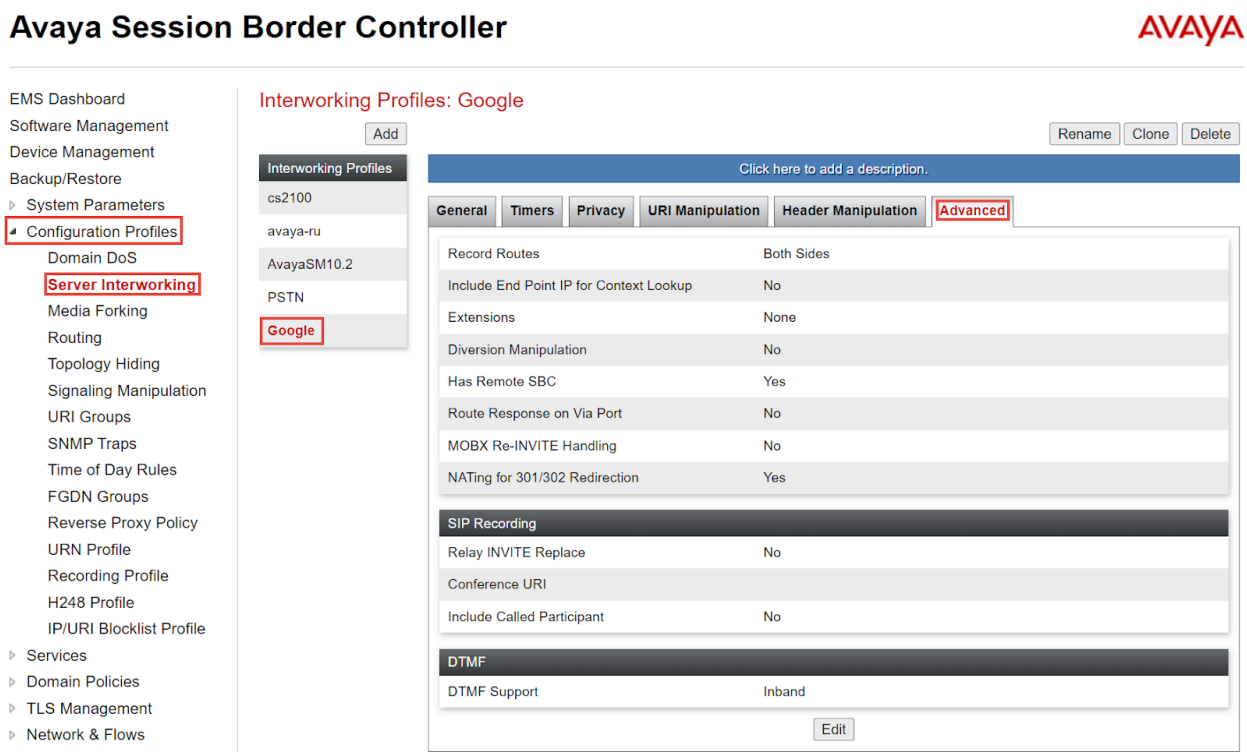

Figure 7: Server Interworking Profile for Google CCAI (Cont.)

Server Interworking for **PSTN Gateway**

- Repeat the same procedure to create the Interworking Profile towards **PSTN Gateway**

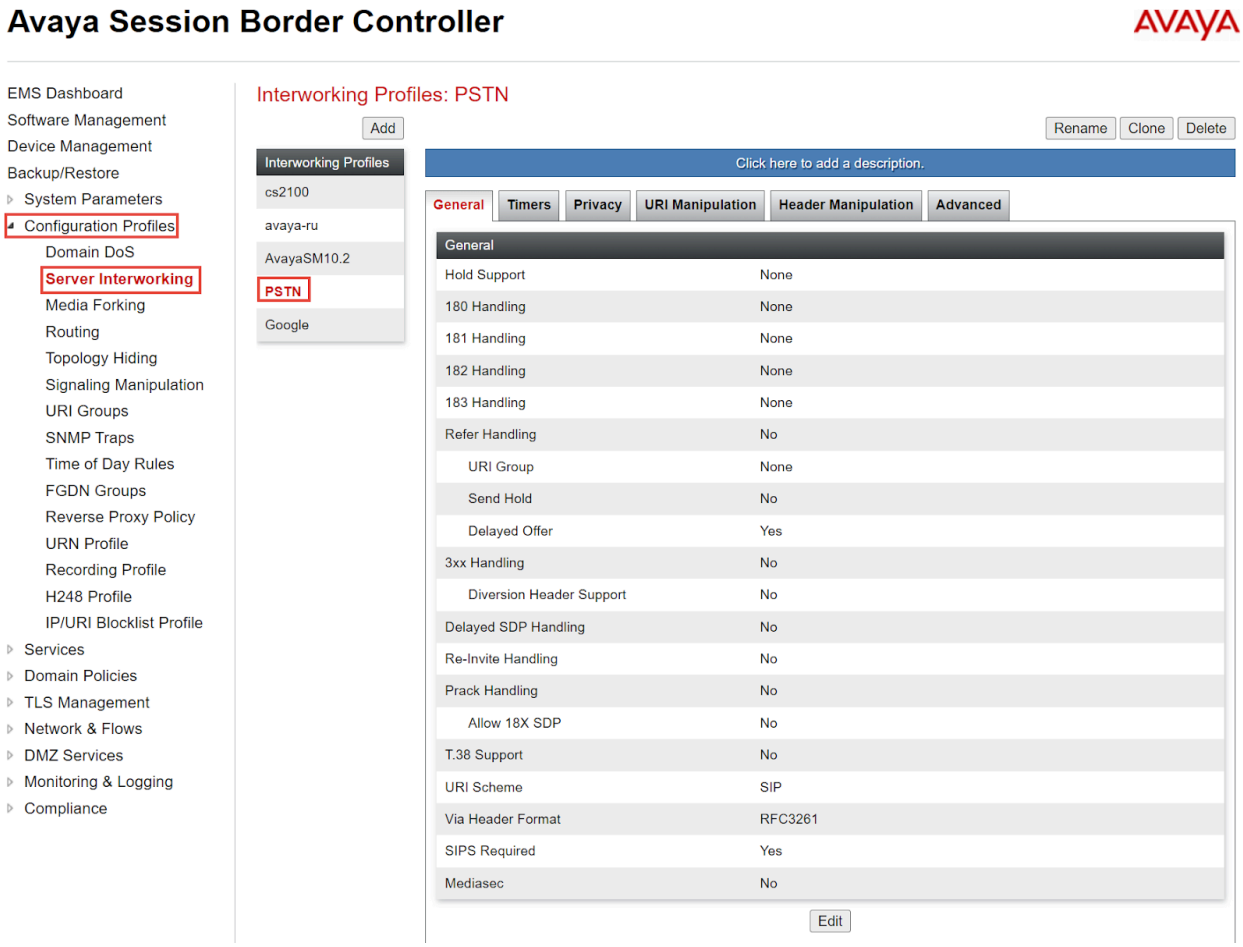

Figure 8: Server Interworking Profile for PSTN Gateway

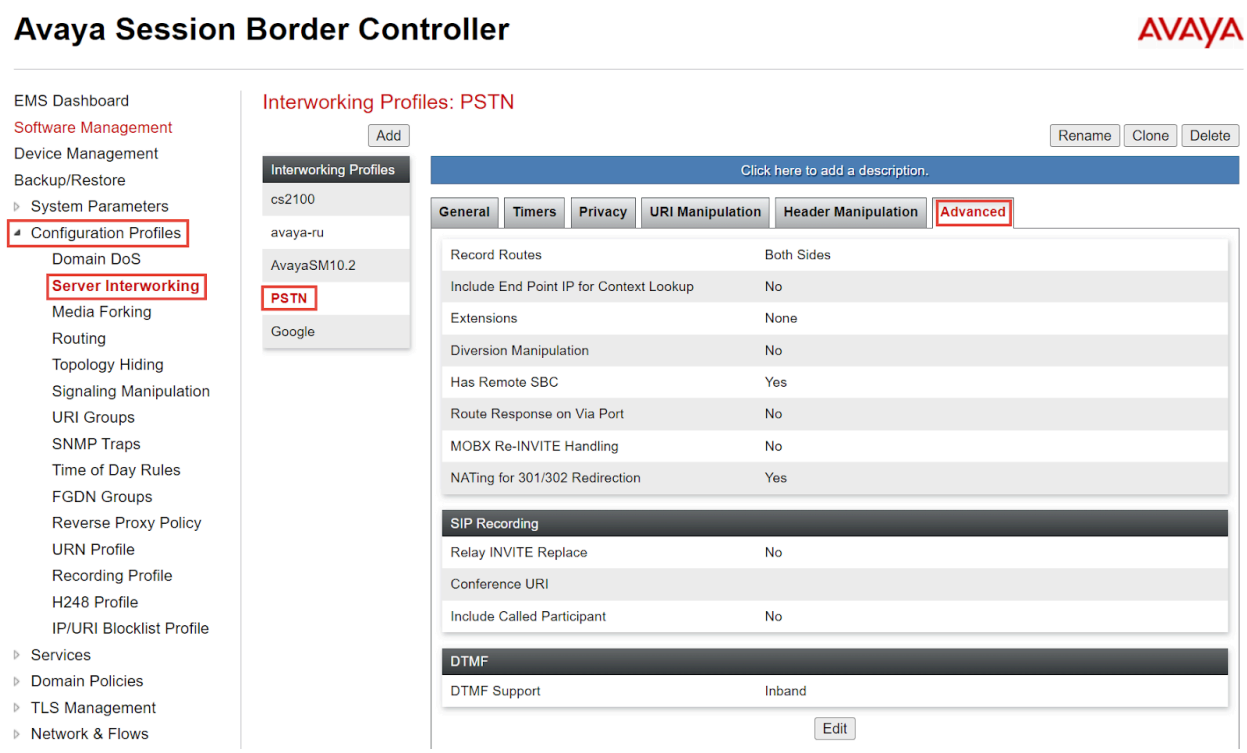

Figure 9: Server Interworking Profile for PSTN Gateway (Cont.)

7.4.3 SIP Servers

SIP Server for **Avaya Aura SM**

- Navigate: **Services > SIP Servers**
- Click **Add**
- Set Profile Name: **AvayaSM10.2**
- Click **Next**

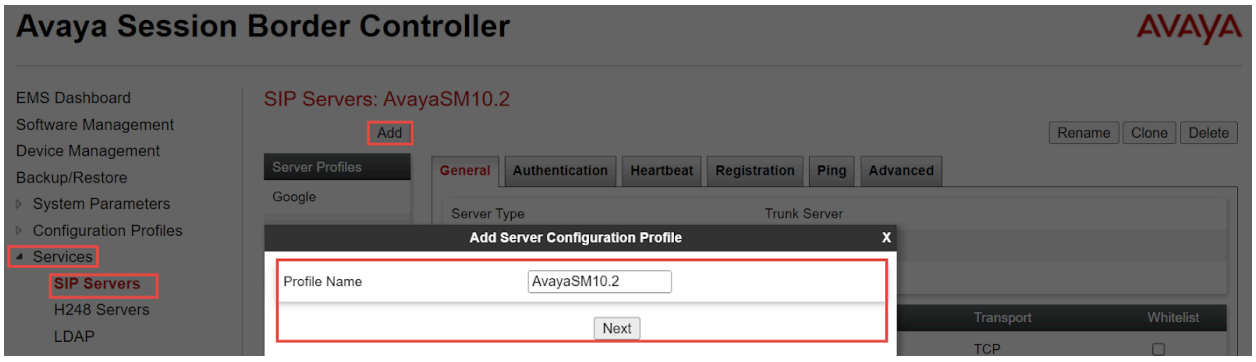

Figure 10: SIP Server for Avaya Aura SM

- Set Server Type: Select Trunk Server from the drop down
- Set IP Address/FQDN/CIDR Range: Enter the Avaya Aura SM IP Address
- Set Port: **5060**
- Set Transport: **TCP**
- Click **Finish**

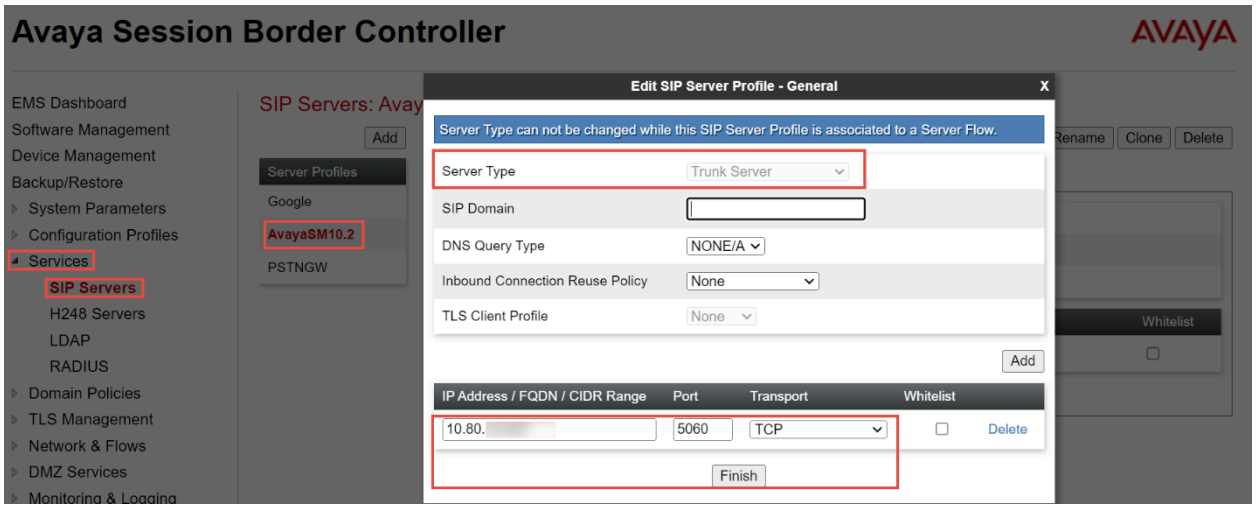

Figure 11: SIP Server for Avaya Aura SM (Cont.)

- Navigate: **Heartbeat** tab
- Set Enable Heartbeat: **Checked**
- Click **Finish**

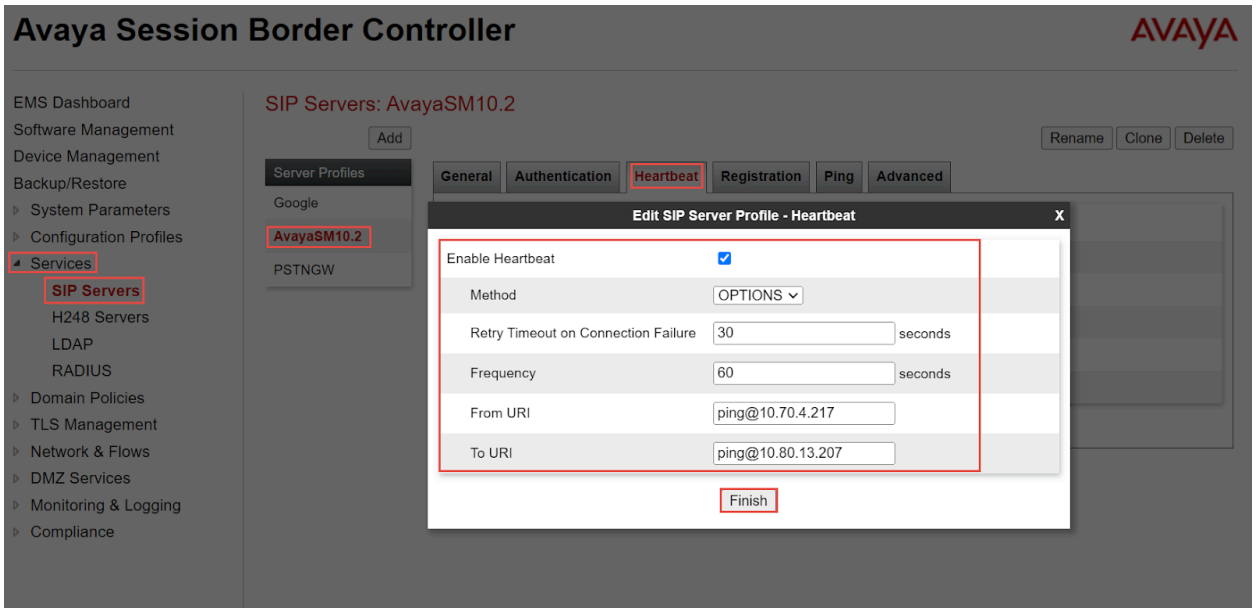

Figure 12: SIP Server for Avaya Aura SM (Cont.)

- Navigate: **Ping** tab
- Set Enable Ping: **Checked**
- Click **Finish**

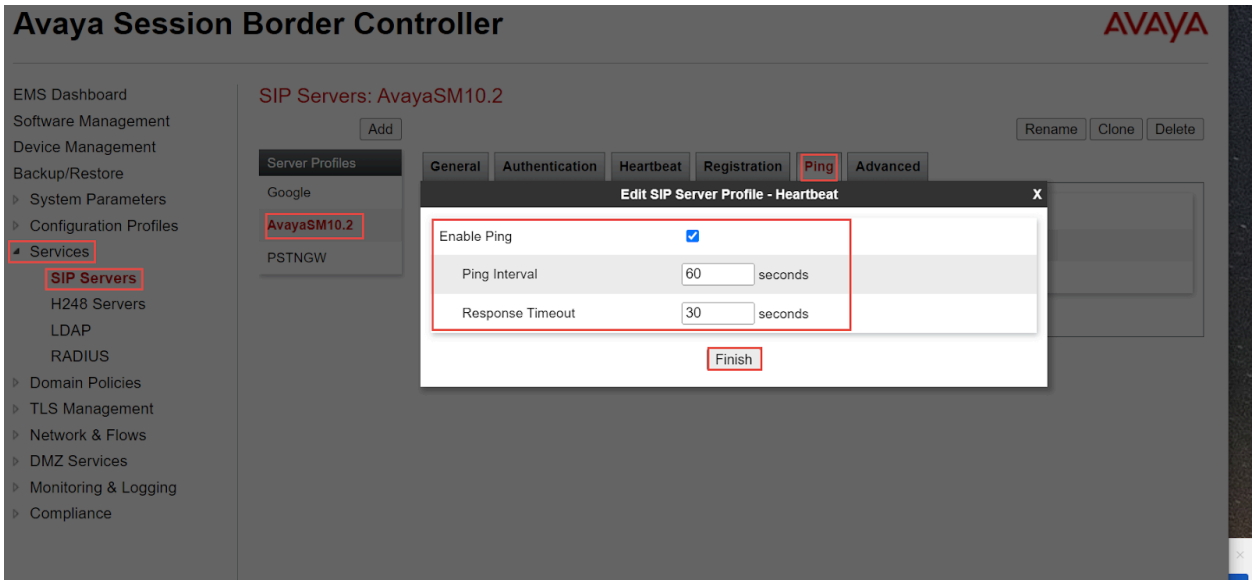

Figure 13: SIP Server for Avaya Aura SM (Cont.)

- Navigate: **Advanced** tab
- Set Enable Grooming: **Checked**
- Set Interworking Profile: Select **AvayaSM10.2**

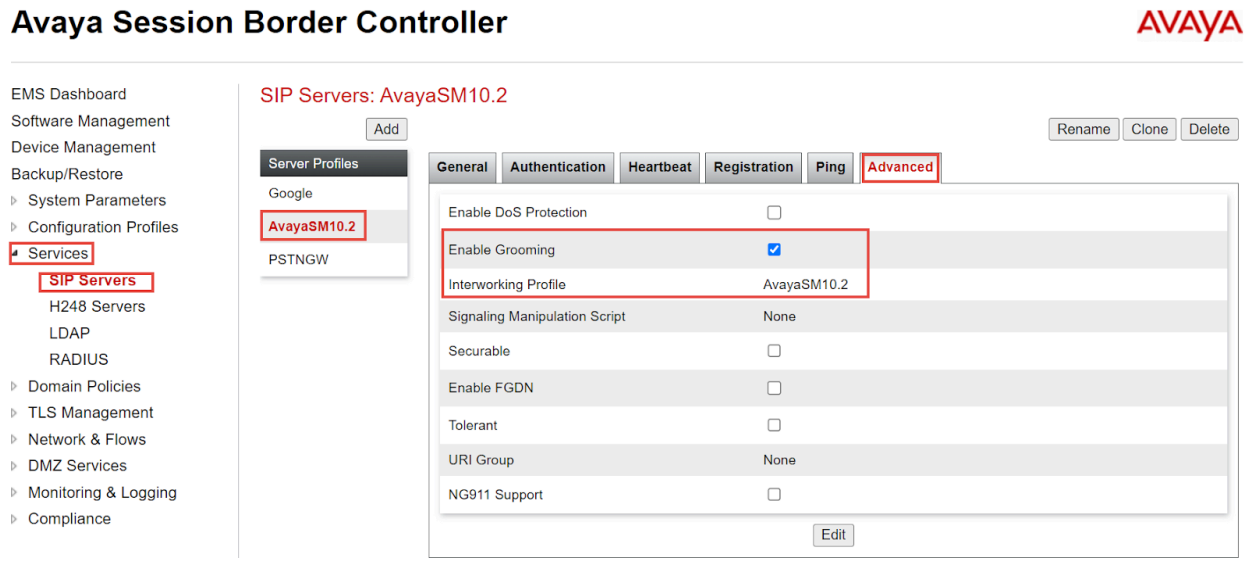

Figure 14: SIP Server for Avaya Aura SM (Cont.)

SIP Server for **Google CCAI**

- Navigate: **Services > SIP Servers**
- Click **Add**
- Set Profile Name: **Google**
- Click **Next**

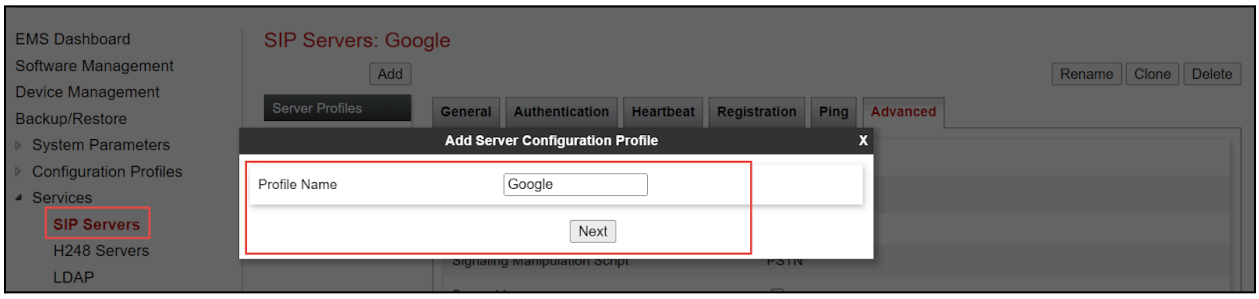

Figure 15: SIP Server for Google CCAI

- Set Server Type: Select Recording Server from the drop down
- Set IP Address/FQDN: Enter Google CCAI FQDN
- Set Port: **5672**
- Set Transport: **TLS**
- Click **Finish**

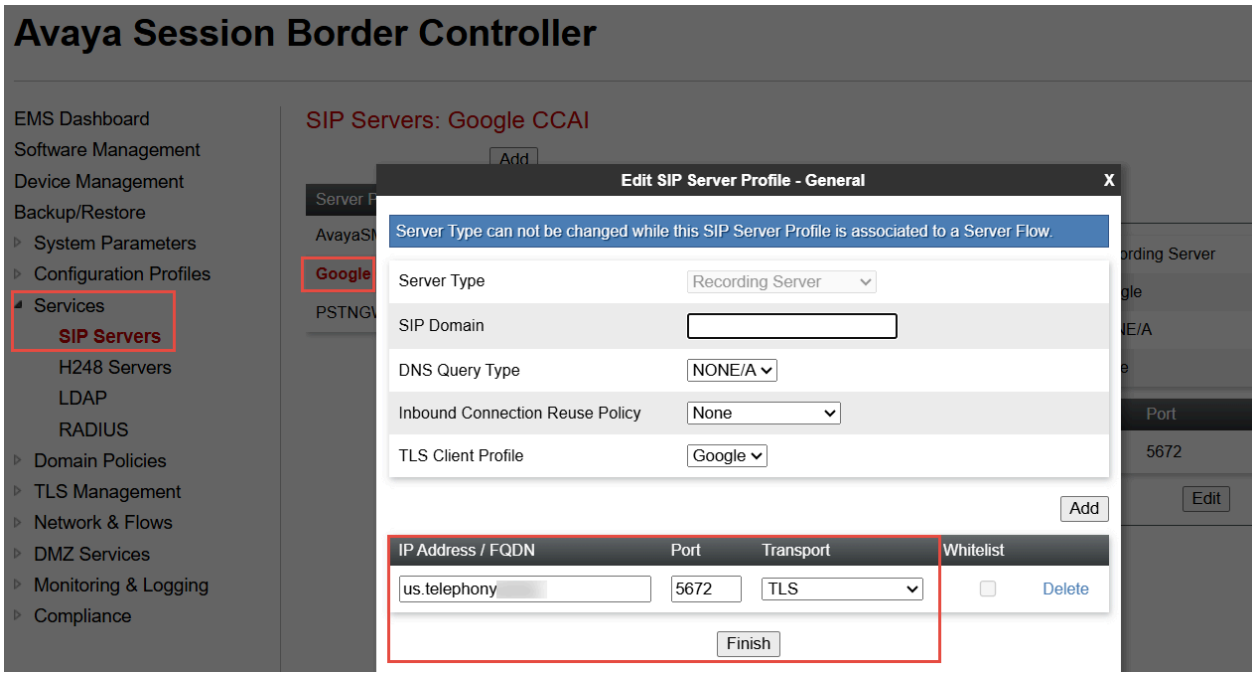

Figure 16: SIP Server for Google CCAI (Cont.)

- Navigate: **Heartbeat** tab
- Set Enable Heartbeat: **Checked**
- Click **Finish**

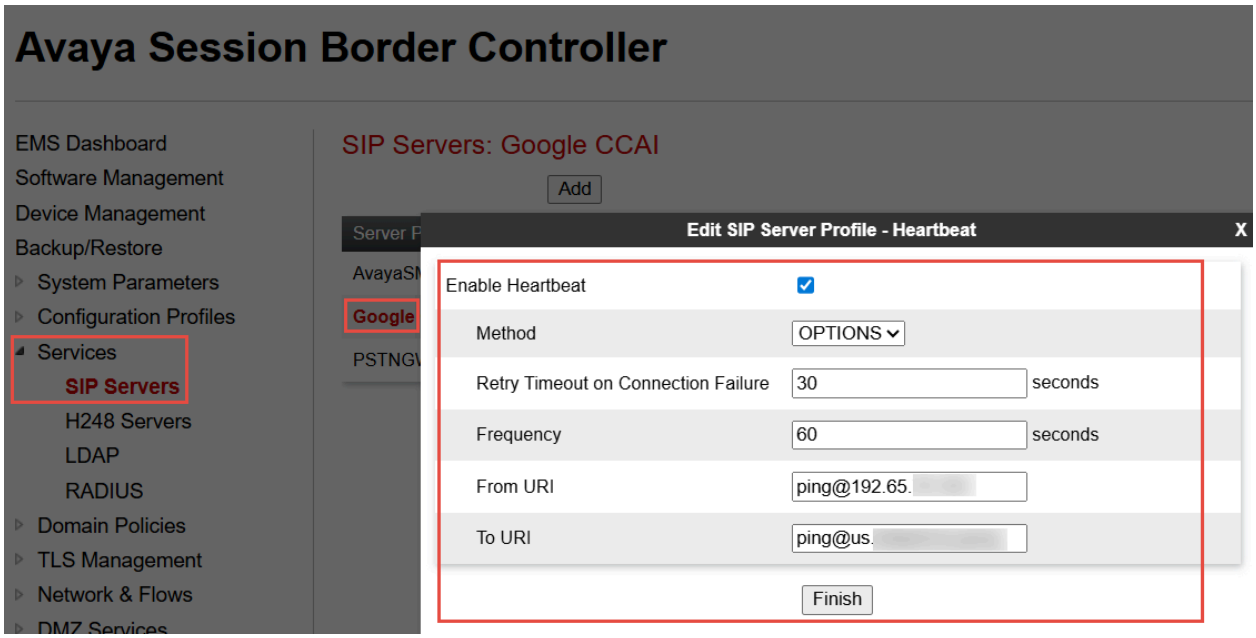

Figure 17: SIP Server for Google CCAI (Cont.)

- Navigate to **Ping** tab
- Set Enable Ping: **Checked**
- Click **Finish**

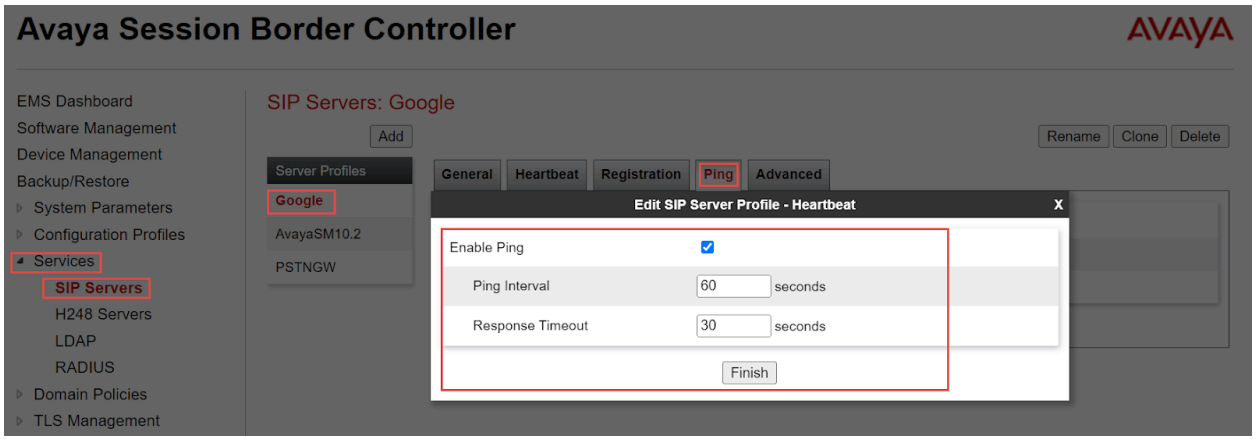

Figure 18: SIP Server for Google CCAI (Cont.)

- Navigate: **Advanced** tab
- Set Enable Grooming: **Checked**
- Set Interworking Profile: Select **Google**
- Click **Finish**

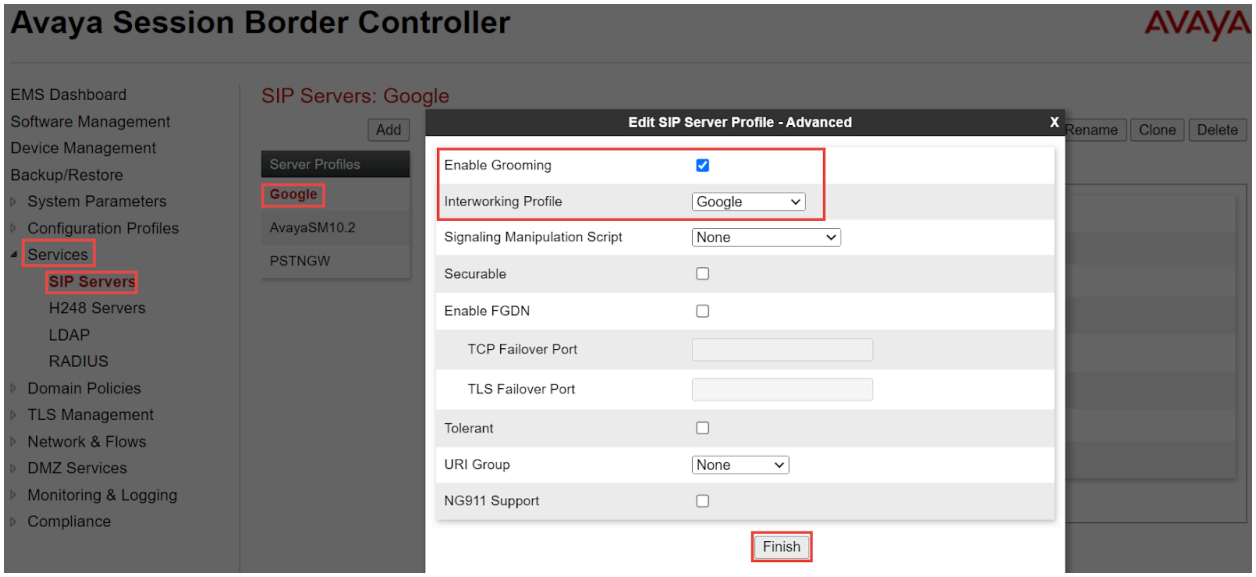

Figure 19: SIP Server for Google CCAI (Cont.)

SIP Server for **PSTN Gateway**

- Navigate: **Services > SIP Servers**
- Click **Add**
- Set Profile Name: **PSTNGW**
- Click **Next**

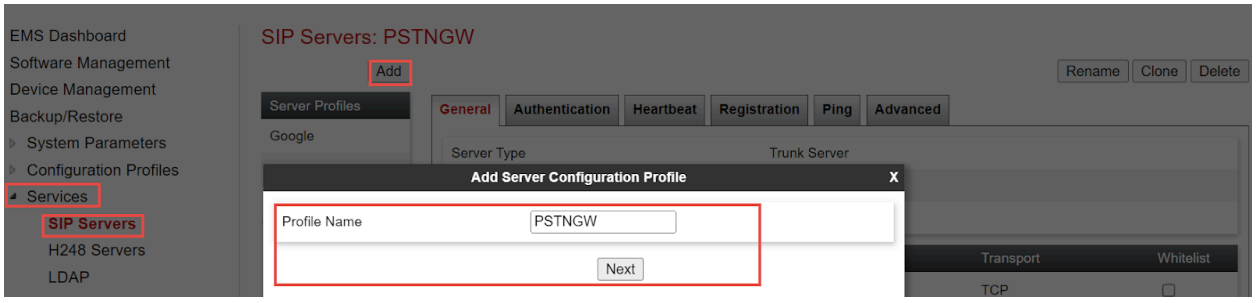

Figure 20: SIP Server for PSTN Gateway

- Set Server Type: Select Trunk Server from the drop down
- Set IP Address/FQDN: Enter the PSTN Gateway IP address.
- Set Port: **5060**
- Set Transport: **TCP**
- Click **Finish**

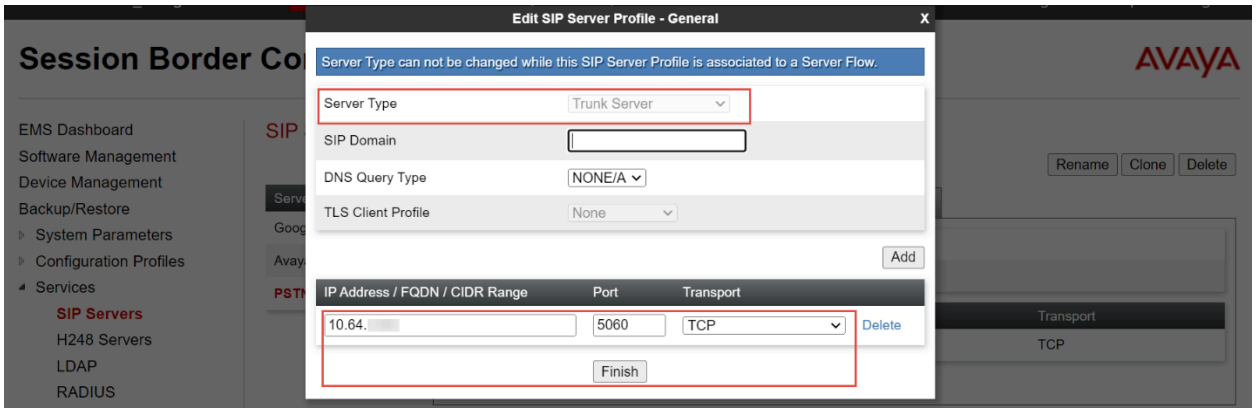

Figure 21: SIP Server for PSTN Gateway (Cont.)

- Navigate: **Heartbeat** tab
- Set Enable Heartbeat: **Checked**
- Click **Finish**

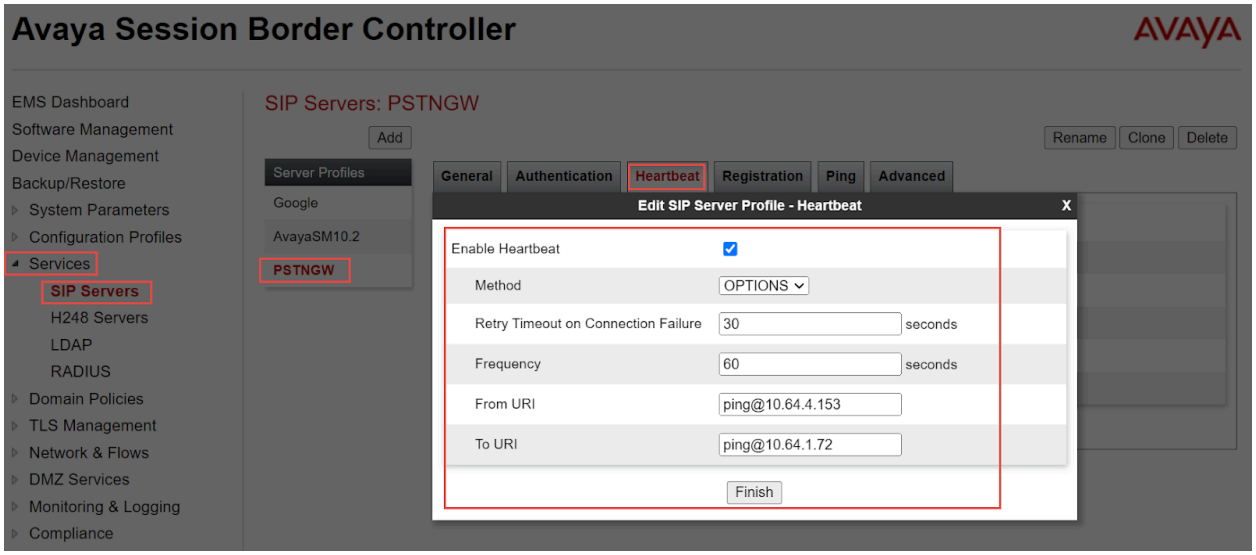

Figure 22: SIP Server for PSTN Gateway (Cont.)

- Navigate: **Ping** tab
- Set Enable Ping: **Checked**
- Click **Finish**

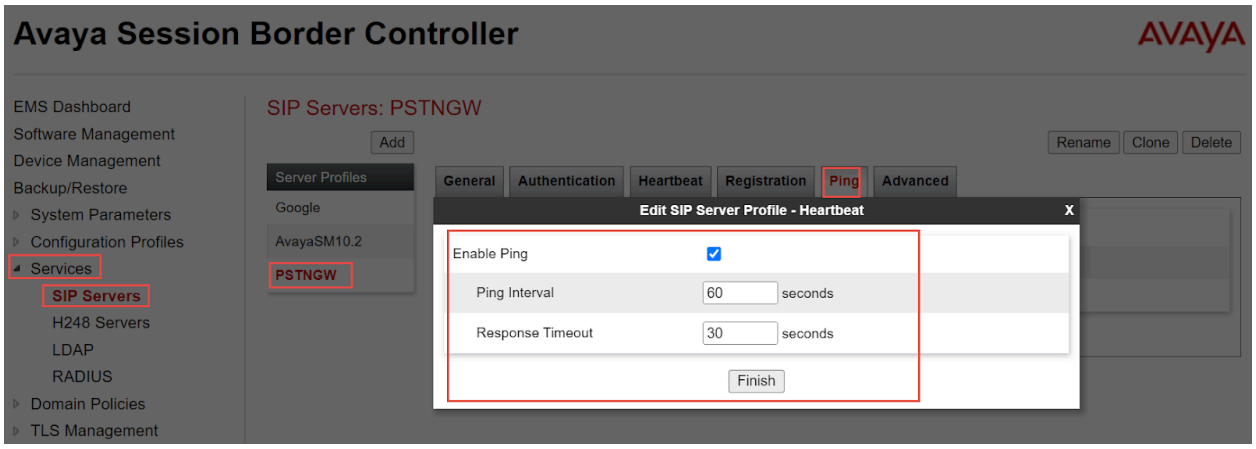

Figure 23: SIP Server for PSTN Gateway (Cont.)

- Navigate: **Advanced** tab
- Set Enable Grooming: **Checked**
- Set Interworking Profile: Select **PSTN**
- Click **Finish**

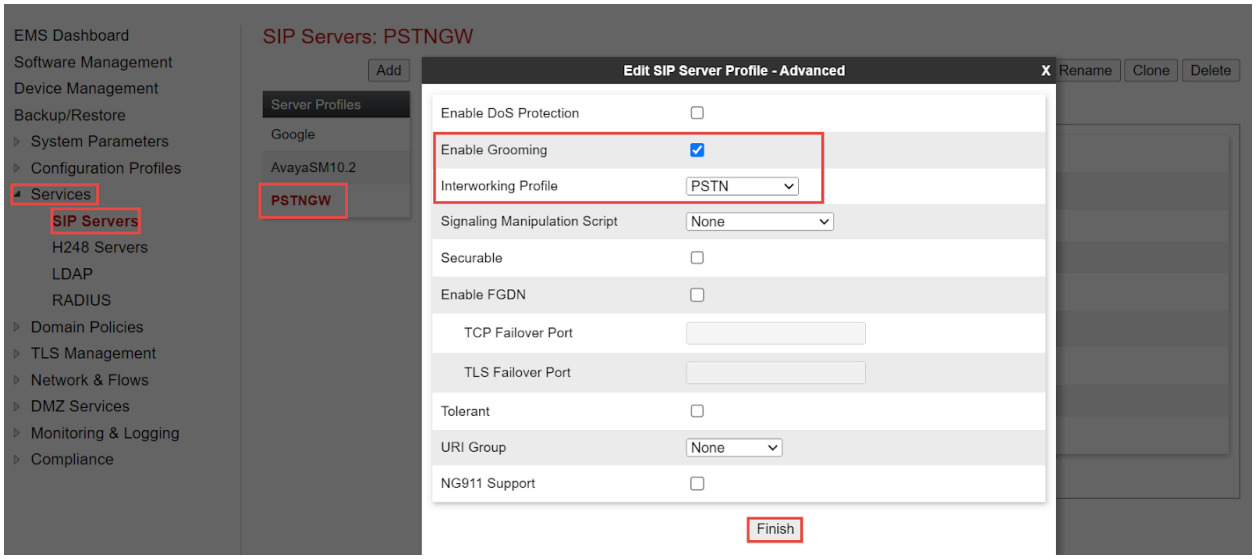

Figure 24: SIP Server for PSTN Gateway (Cont.)

7.4.4 Topology Hiding

Topology Hiding profile for **Google**

- Topology Hiding profiles are added for Google CCAI to overwrite and hide certain headers
- Navigate: **Configuration Profiles > Topology Hiding**
- Click **Add**
- Set Profile Name: **Google CCAI**
- Click **Next**

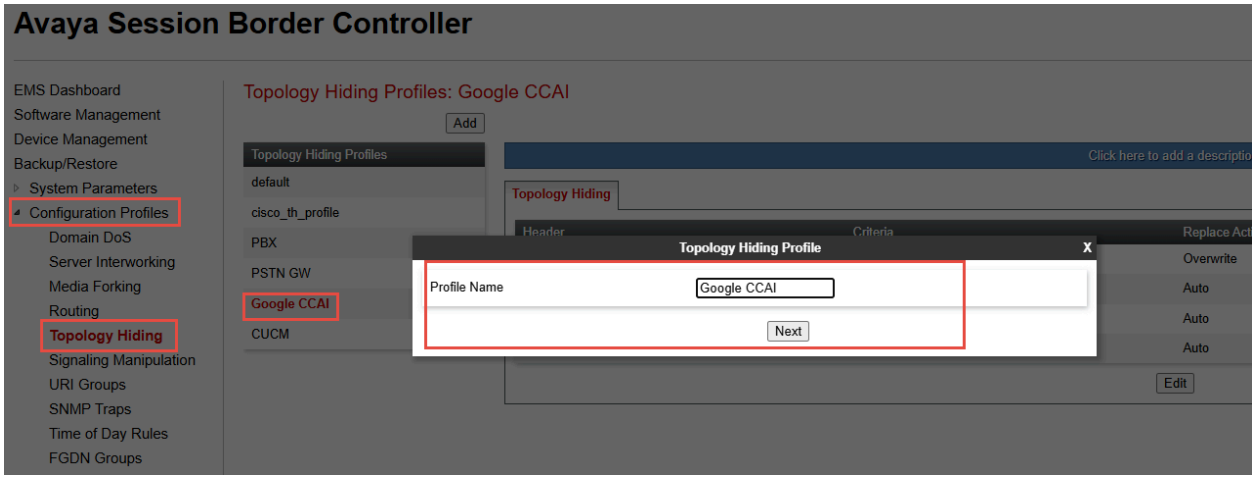

Figure 25: Topology Hiding for Google CCAI

- Select the newly created profile **Google** and Click **Edit**
- Overwrite Value: Replace the **From Header** with Google CCAI Facing Public IP
- Click **Finish**

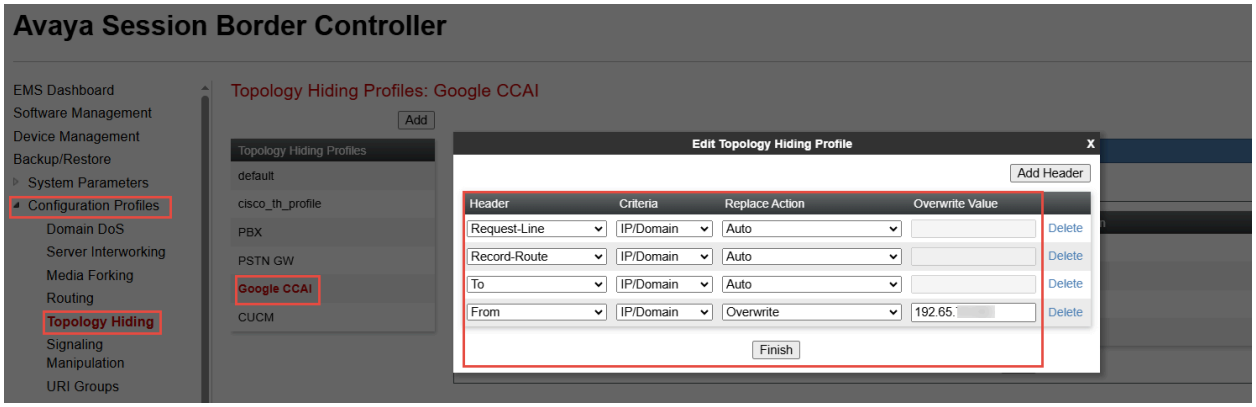

Figure 26: Topology Hiding for Google CCAI (Cont.)

7.4.5 Routing

Routing for **Avaya Aura SM**

- Navigate: **Configuration Profiles > Routing**
- Click **Add**
- Set Profile Name: **AvayaSM10.2**
- Click **Next**

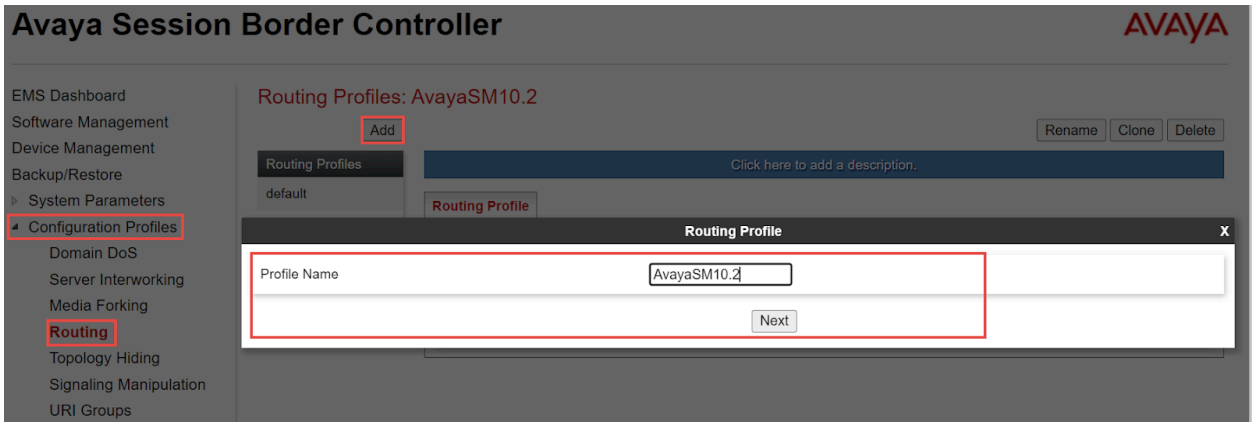

Figure 27: Routing for Avaya Aura SM

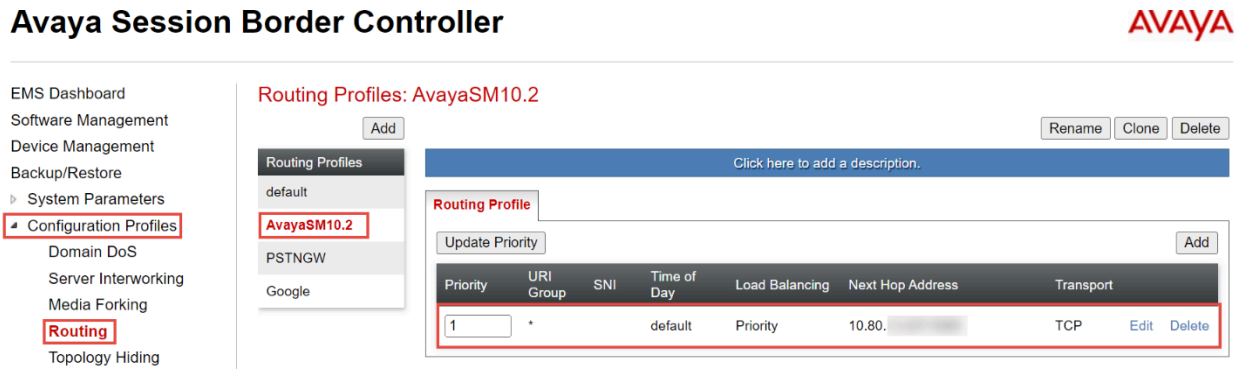

Figure 28: Routing for Avaya Aura SM (Cont.)

- At Routing Profile Window, Click **Add**
- Set Priority/Weight: **1**
- Select **SIP Server Profile, Next Hop Address** from the drop-down menu
- Click **Finish**

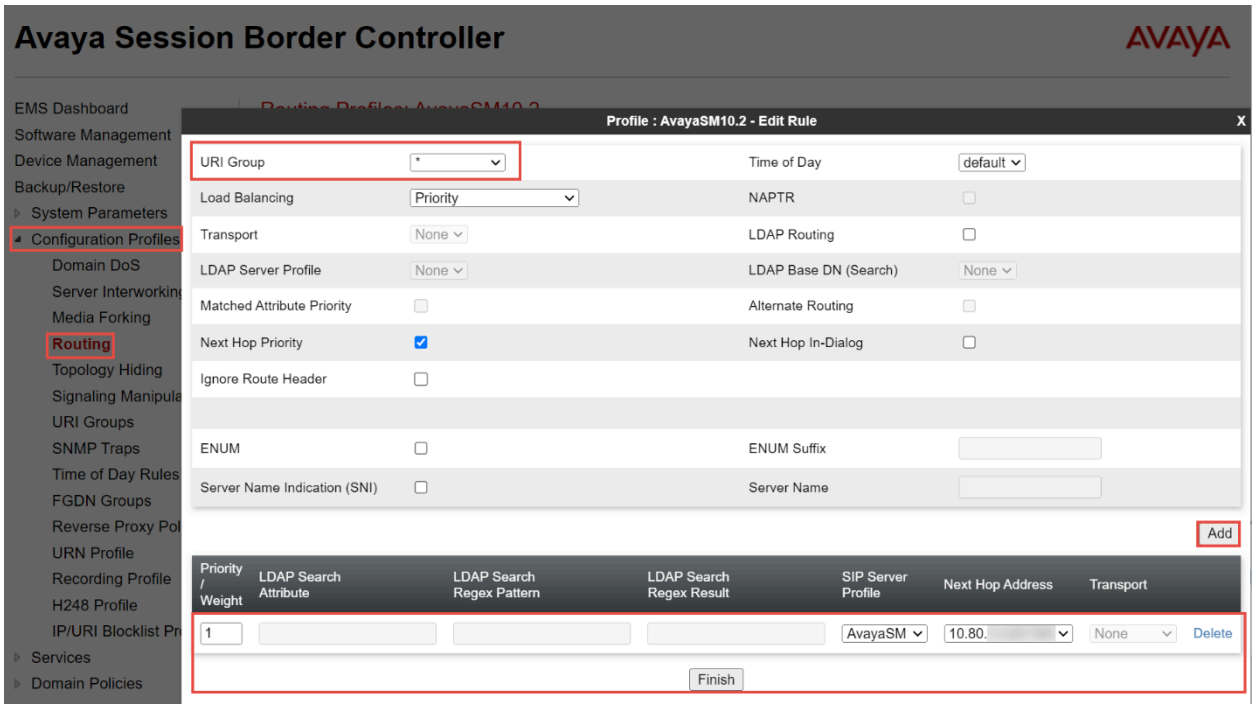

Figure 29: Routing for Avaya Aura SM (Cont.)

Routing for **PSTN Gateway**

- Navigate: **Configuration Profiles > Routing**
- Click **Add**
- Set Profile Name: **PSTNGW**
- Click **Next**

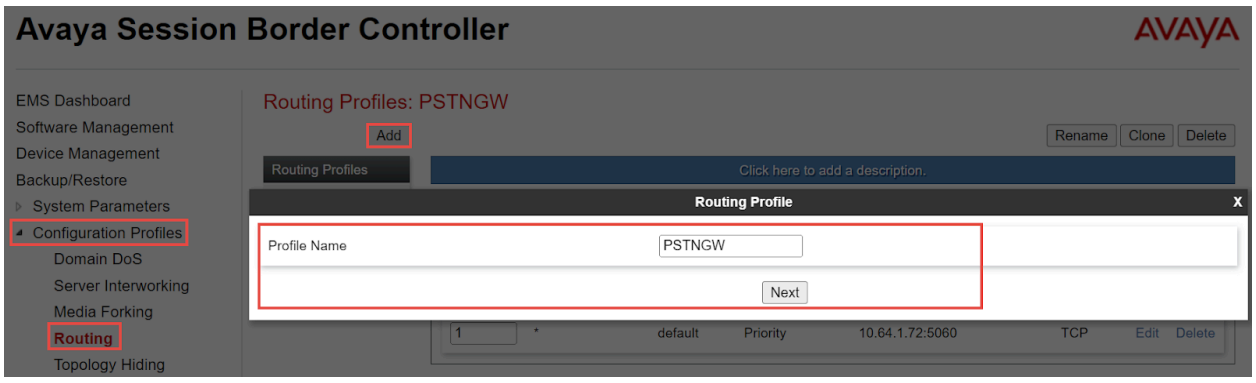

Figure 30: Routing for PSTN Gateway

- At Routing Profile Window, Click **Add**
- Set Priority/Weight: **1**
- Select **SIP Server Profile, Next Hop Address** from the drop-down menu
- Click **Finish**

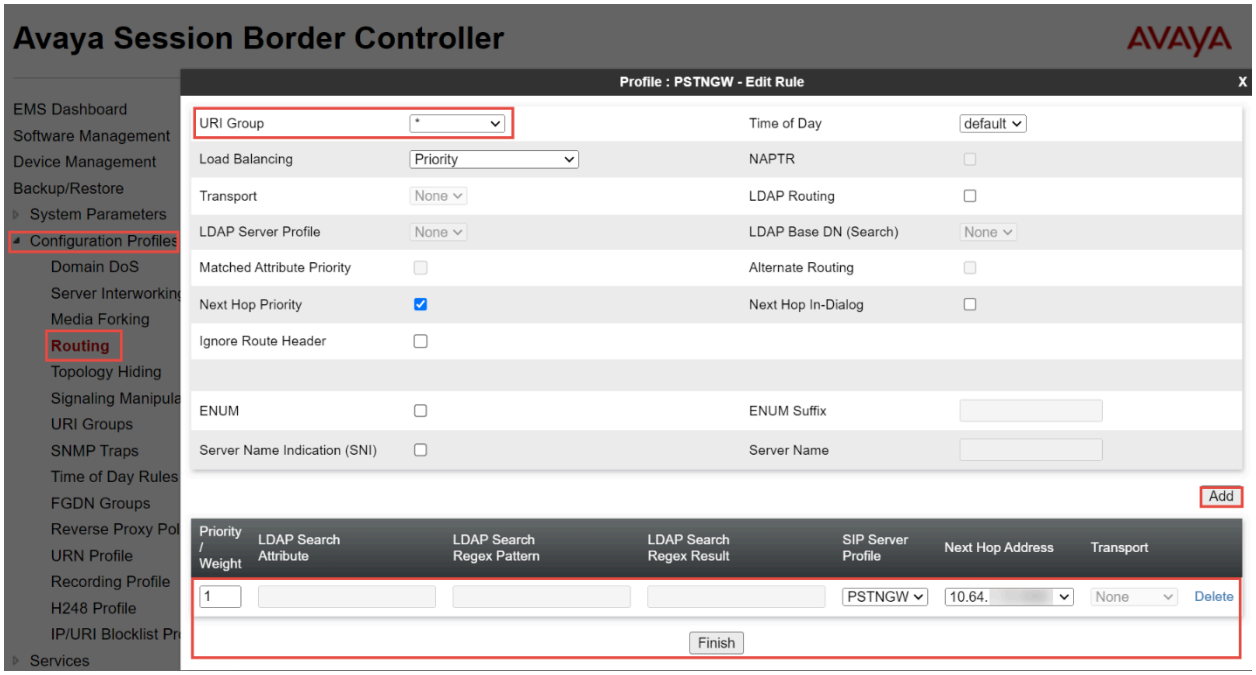

Figure 31: Routing for PSTN Gateway (Cont.)

Routing for **Google CCAI**

- Navigate: **Configuration Profiles > Routing**
- Click **Add**
- Set Profile Name: **Google**
- Click **Next**

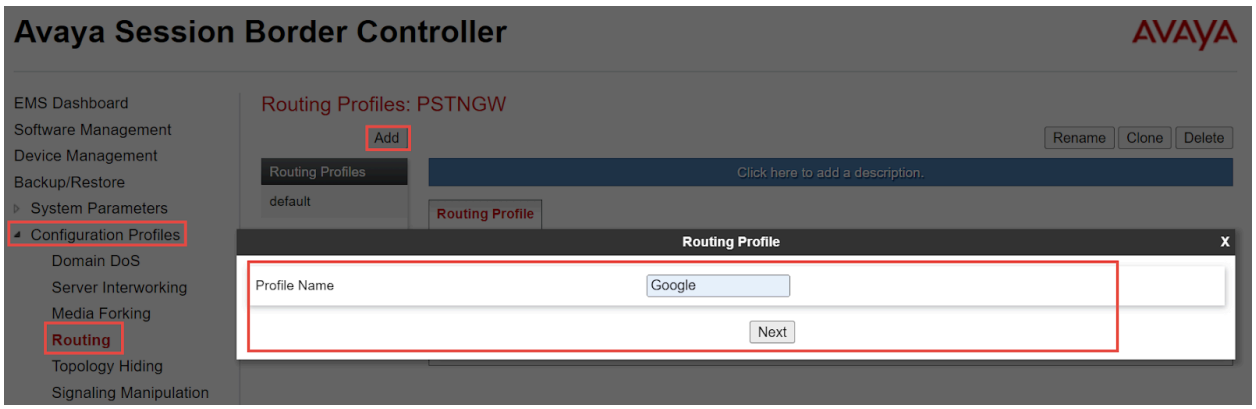

Figure 32: Routing for Google CCAI

- At Routing Profile Window, Click **Add**
- Set Priority/Weight: **1**
- Select **SIP Server Profile, Next Hop Address** from the drop-down menu
- Click **Finish**

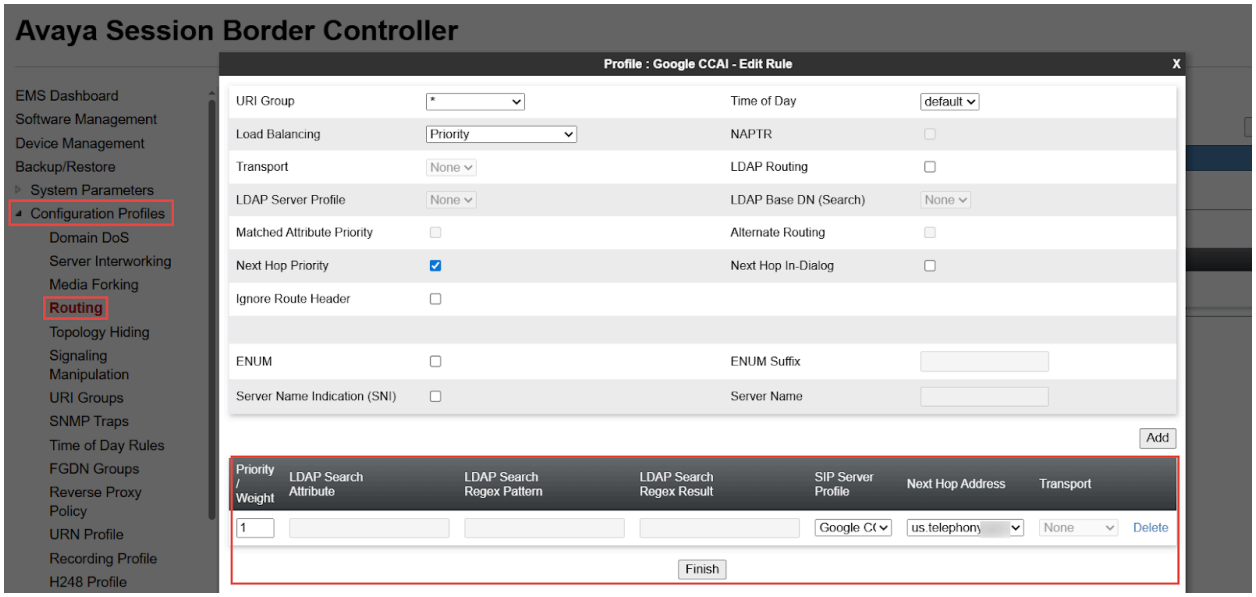

Figure 33: Routing for Google CCAI (Cont.)

7.4.6 Recording Profile

- Navigate: **Configuration> Recording Profile**
- Click **Add**
- Set Profile Name: **Google_RP**
- Click **Next**

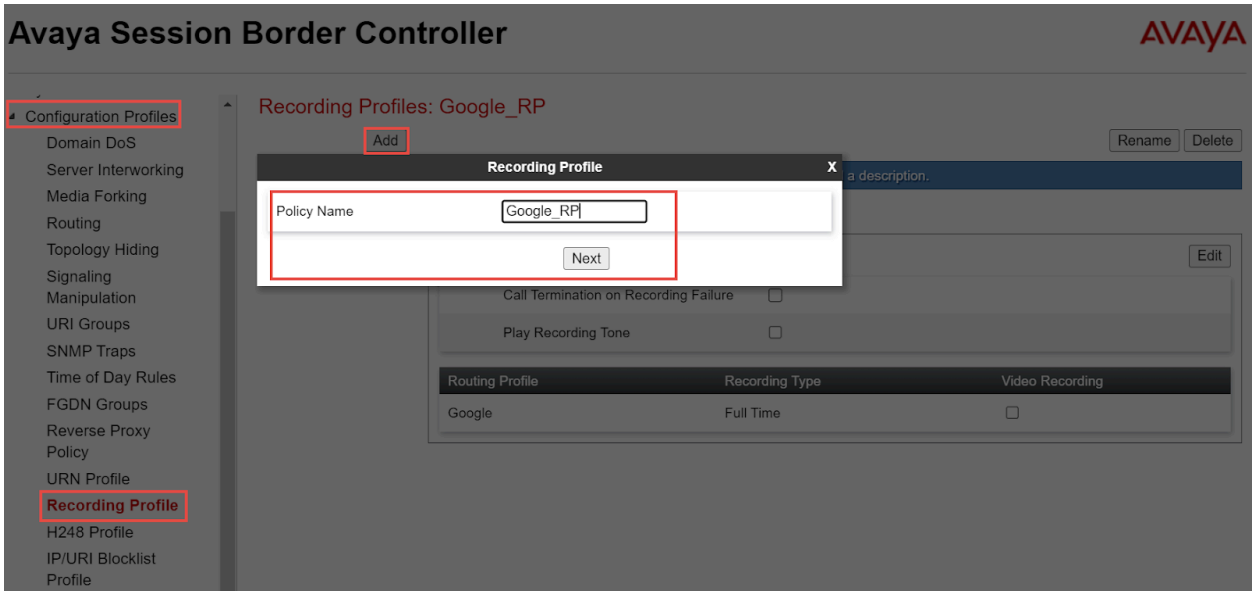

Figure 34: Recording Profile for Google CCAI

- Set Routing Profile: Select **Google**
- Set Recording Type: Select **Full Time** from the dropdown
- Click **Finish**

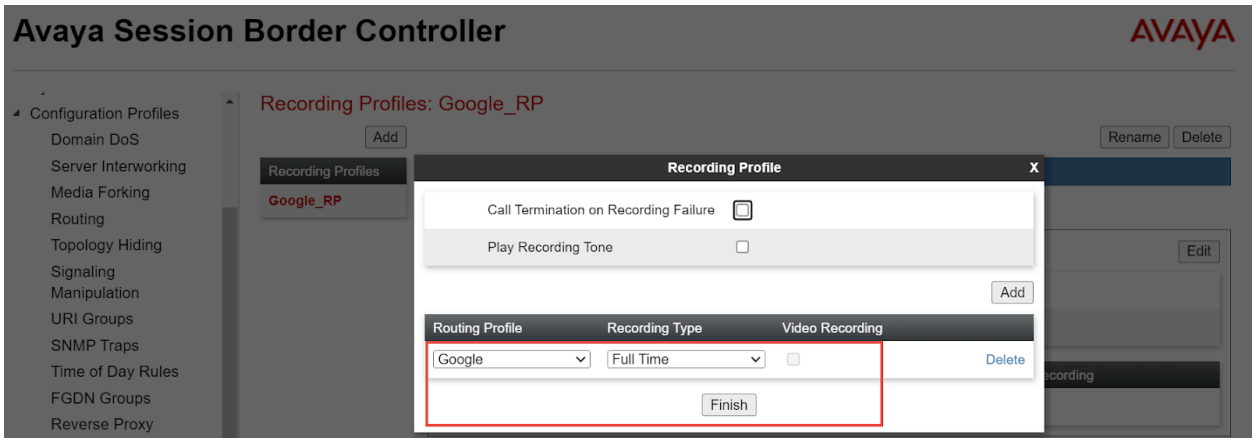

Figure 35: Recording Profile for Google CCAI (Cont.)

7.4.7 Session Policies

- Navigate: **Domain Policies > Session Policies**
- Select default under Session Policies, Click **Clone**
- Set Profile Name: **Google_SP**
- Click **Next**

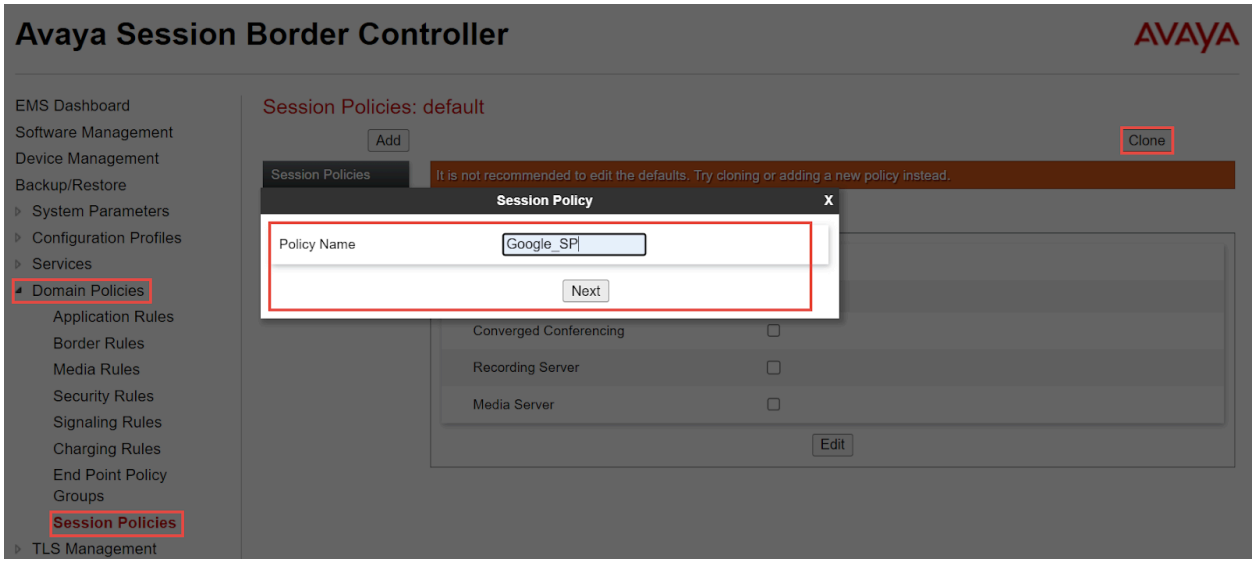

Figure 36: Session Policies for Google CCAI

- Media Anchoring: **Checked**
- Recording Server: **Checked**
- Set Routing Profile: Select the route profile **Google_RP**
- Click **Finish**

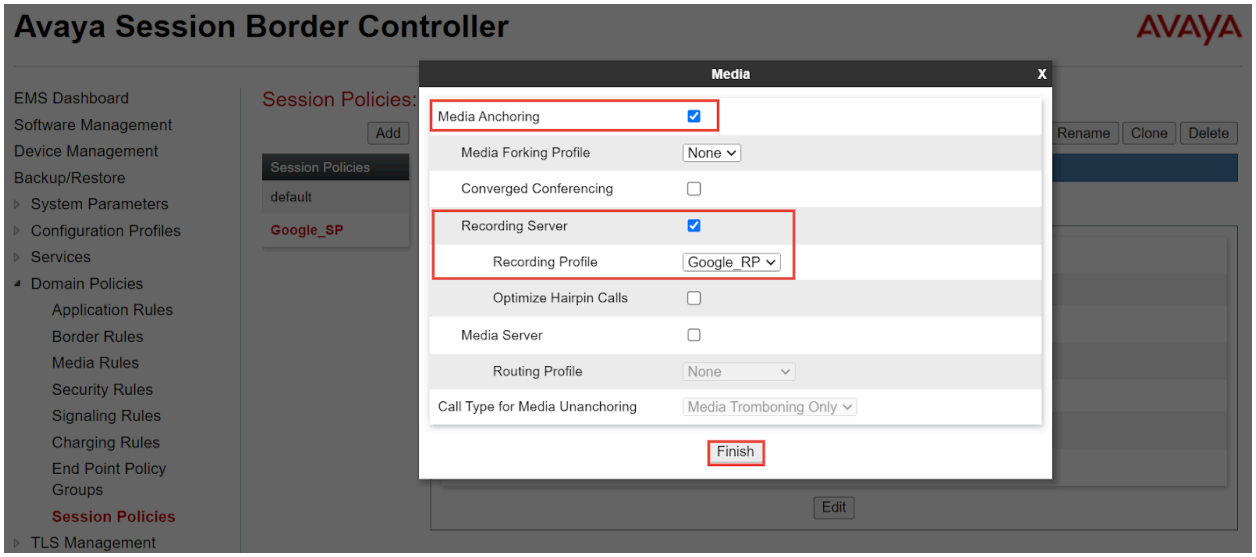

Figure 37: Session Policies for Google CCAI (Cont.)

7.4.8 Session Flows

- Navigate: **Network and Flows> Session Flows**
- Click **Add**
- Set Name: **Google_SF**
- Select Session Policy: **Google_SP**
- Click **Finish**

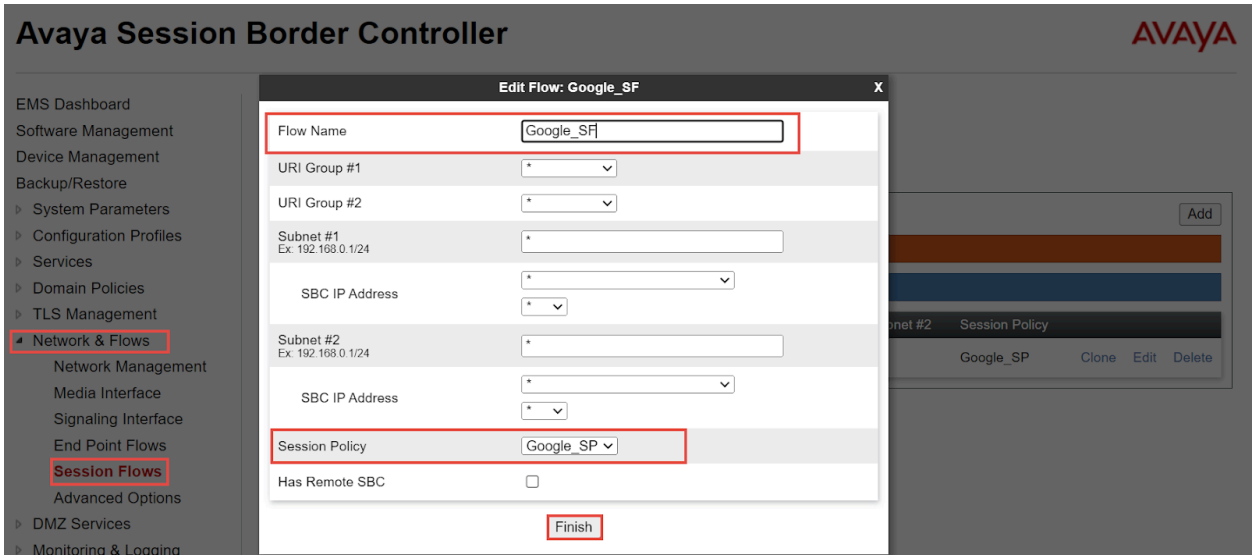

Figure 38: Session flow for Google CCAI

7.4.9 Signaling Manipulation

- Navigate: **Configuration Profiles > Signaling Manipulation**
- Click **Add**
- Title: **Google**
- Click **Save**
- Below sigma script is created to add **Call-Info** header towards Google CCAI with the Dialog Flow API request along with the Conversation ID.
- Avaya signaling manipulation does not allow to add double slash (http://) in the manipulation, hence "**&slash"** is added to the **%baseURI** as shown below. Later "**&slash"** is replaced with symbol **"/"** using manipulations.
- **%baseUri** value provided below is a reference value. Project name("**ccai-38XXXXXconversations"**) present in the call-info header will vary according to the project created by user. **Ab_** is just an identifier, you can use any values which matches the regex pattern requirement of call info header.

```
within session "all"
{
   act on request where %DIRECTION="OUTBOUND" and
%ENTRY_POINT="POST_ROUTING" and %METHOD="INVITE"
   {
      %aor = %HEADERS["Call-ID"][1];
      %baseUri =
"<http:&slash/dialogflow.googleapis.com/v2beta1/projects/ccai-389811/conversations/
Sr_";
      append( %baseUri, %aor);
       %newUri1 = ">;purpose=Goog-ContactCenter-Conversation";
      append( %baseUri, %newUri1);
      %HEADERS["Call-Info"][1] = %baseUri;
      %HEADERS["Call-Info"][1].URI.regex_replace("&slash","/");
      //%HEADERS["Request_Line"][1].regex_replace("+1361400XXXX +1361400XXXX ",
"+1361400XXXX ");
      %HEADERS["Request_Line"][1].URI.USER.regex_replace("(.*)", "+1361400XXXX ");
      %HEADERS["TO"][1].URI.USER.regex_replace("^.........", "+1361400XXXX ");
      %HEADERS["Allow"][1].regex_replace(", UPDATE,", "");
      //%HEADERS["Request_Line"][1].regex_replace("+1361400XXXX +1361400XXXX ",
"+1361400XXXX ");
   }
   act on request where %DIRECTION="OUTBOUND" and
%ENTRY_POINT="POST_ROUTING" and %METHOD="ACK"
   {
      //%HEADERS["Allow"][1].regex_replace(", UPDATE,", "");
      %HEADERS["TO"][1].URI.USER.regex_replace("^.........", "+1361400XXXX ");
      }
   act on request where %DIRECTION="OUTBOUND" and
%ENTRY_POINT="POST_ROUTING" and %METHOD="UPDATE"
   {
      //%HEADERS["Allow"][1].regex_replace(", UPDATE,", "");
      %HEADERS["TO"][1].URI.USER.regex_replace("^.........", "+1361400XXXX");
```

```
                %HEADERS["Request_Line"][1].regex_replace(";transport=udp", "");
                %HEADERS["Content-Type"][1].regex_replace("application/rs-metadata",
            "application/rs-metadata+xml");

                }
    }
```

```
within session "all"
{
    act on request where %DIRECTION="OUTBOUND" and  %ENTRY_POINT="POST_ROUTING" and %METHOD="INVITE"
    {
        %aor = %HEADERS["Call-ID"][1];
        %baseUri = "<http:&slash/dialogflow.googleapis.com/v2beta1/projects/ccai-389811/conversations/Sr_";
        append( %baseUri, %aor);
          %newUri1 = ">;purpose=Goog-ContactCenter-Conversation";
        append( %baseUri, %newUri1);
        %HEADERS["Call-Info"][1] = %baseUri;
        %HEADERS["Call-Info"][1].URI.regex_replace("&slash","/");
        //%HEADERS["Request_Line"][1].regex_replace("+13614003   4+136140    ", "+1361400    ");
        %HEADERS["Request_Line"][1].URI.USER.regex_replace("(.*)", "+1361400    ");
        %HEADERS["TO"][1].URI.USER.regex_replace("^.........", "+1361400    ");
        %HEADERS["Allow"][1].regex_replace(", UPDATE,", "");
        //%HEADERS["Request_Line"][1].regex_replace("+136140    +1361400    ", "+136140    4");
    }
    act on request where %DIRECTION="OUTBOUND" and %ENTRY_POINT="POST_ROUTING" and %METHOD="ACK"
    {
        //%HEADERS["Allow"][1].regex_replace(", UPDATE,", "");
        %HEADERS["TO"][1].URI.USER.regex_replace("^.........", "+136140    ");
        }
    act on request where %DIRECTION="OUTBOUND" and %ENTRY_POINT="POST_ROUTING" and %METHOD="UPDATE"
    {
        //%HEADERS["Allow"][1].regex_replace(", UPDATE,", "");
        %HEADERS["TO"][1].URI.USER.regex_replace("^.........", "+136140    ");
        %HEADERS["Request_Line"][1].regex_replace(";transport=udp", "");
        %HEADERS["Content-Type"][1].regex_replace("application/rs-metadata", "application/rs-metadata+xml");

        }
    }
```

Figure 39: Signaling Manipulation - Google CCAI

Below Signaling manipulation is particularly used for Participant Label test case:

```
        within session "all"
        {
          act on request where %DIRECTION="OUTBOUND" and  %ENTRY_POINT="POST_ROUTING"
        and %METHOD="INVITE"
          {
            %aor = %HEADERS["Call-ID"][1];
            %baseUri =
        "<http:&slash/dialogflow.googleapis.com/v2beta1/projects/ccai-389811/conversations/Sr_";
            append( %baseUri, %aor);
            %newUri1 =
        "?roles=HUMAN_AGENT,END_USER>;purpose=Goog-ContactCenter-Conversation";
            append( %baseUri, %newUri1);
            %HEADERS["Call-Info"][1] = %baseUri;
            %HEADERS["Call-Info"][1].URI.regex_replace("&slash","/");
            %HEADERS["Request_Line"][1].URI.USER.regex_replace("(^.........)", "+1314944XXXX");
            %HEADERS["TO"][1].URI.USER.regex_replace("^.........", "+1314944XXXX");
                %HEADERS["FROM"][1].URI.USER.regex_replace("^.........", "+214550XXXX");
            %HEADERS["Allow"][1].regex_replace(", UPDATE,", "");
          }
          act on request where %DIRECTION="OUTBOUND" and %ENTRY_POINT="POST_ROUTING"
        and %METHOD="ACK"
          {
            //%HEADERS["Allow"][1].regex_replace(", UPDATE,", "");
```

```
        %HEADERS["TO"][1].URI.USER.regex_replace("^.....", "+1314944XXXX");
        }
    act on request where %DIRECTION="OUTBOUND" and %ENTRY_POINT="POST_ROUTING"
and %METHOD="UPDATE"
    {
        //%HEADERS["Allow"][1].regex_replace(", UPDATE,", "");
        %HEADERS["TO"][1].URI.USER.regex_replace("^.....", "+13149445XXXX");
        %HEADERS["Request_Line"][1].regex_replace(";transport=udp", "");
        //%HEADERS["Content-Type"][1].regex_replace("application/rs-metadata",
"application/rs-metadata+xml");
        }
    }
```

7.4.10 Signaling Rules

- Configure Navigate: **Domain Policies > Signaling Rules**
- Select default under Signaling Rules, Click **Clone**
- Set Rule Name: **Avaya SM**
- Click **Finish**

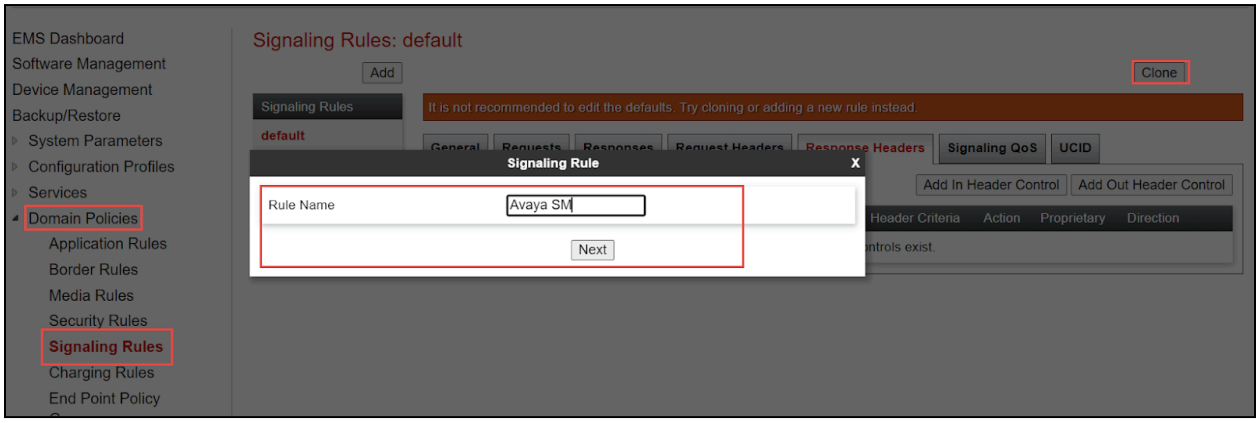

Figure 40: Signaling Rules for Avaya Aura SM

- Select the newly cloned **Signaling Rule Avaya_SM**, under tab **Request Headers**, Click Add In Header Control
- Set Proprietary Request Header: **Checked**
- Set Header Name: **AV-Global-Session-ID**
- Set Method Name: Select ALL from the drop down
- Set Header Criteria: Forbidden
- Set Presence Action: Remove header is selected from the drop down
- Click **Finish**

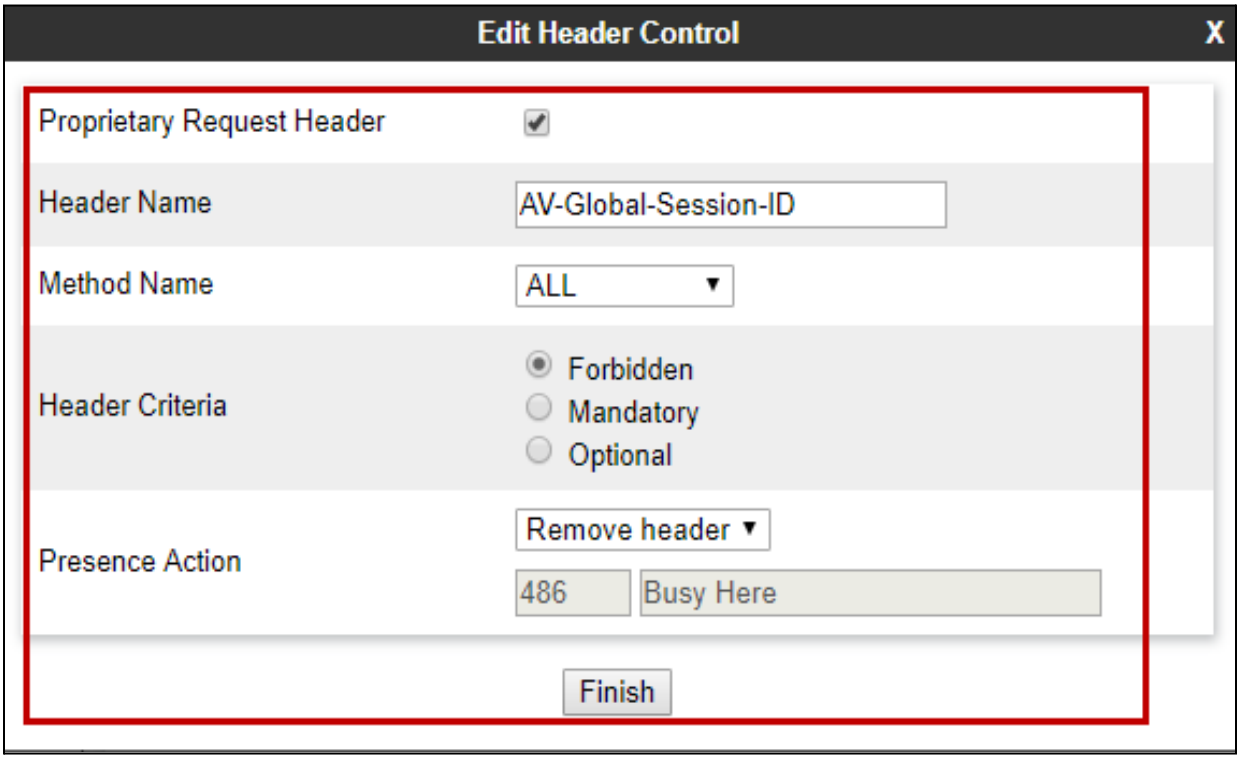

Figure 41: Signaling Rules for Avaya Aura SM (Cont.)

- Repeat the same steps for all other required headers

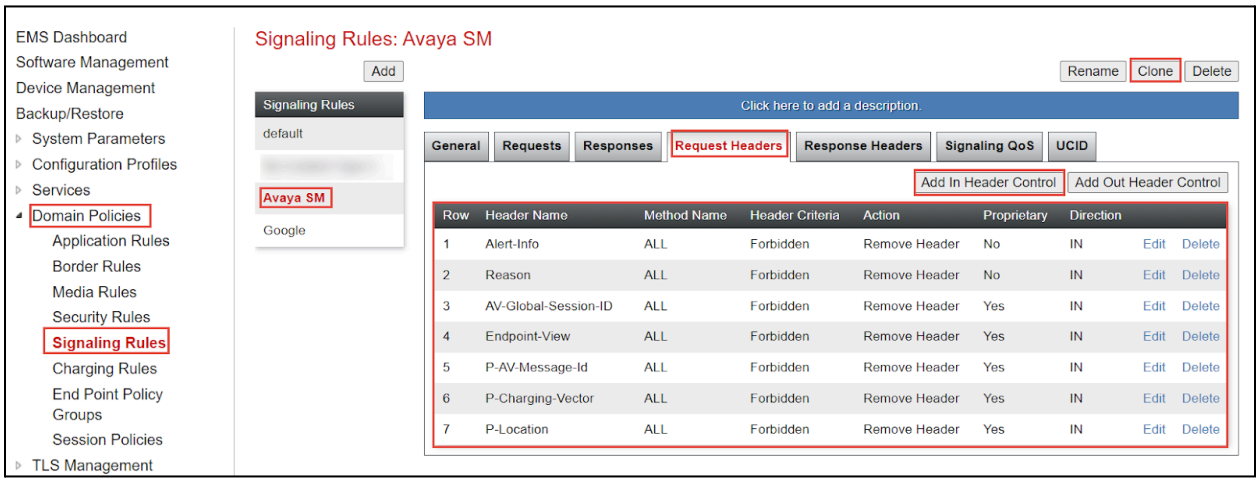

Figure 42: Signaling Rules for Avaya Aura SM (Cont.)

- Repeat the same steps for Response Headers

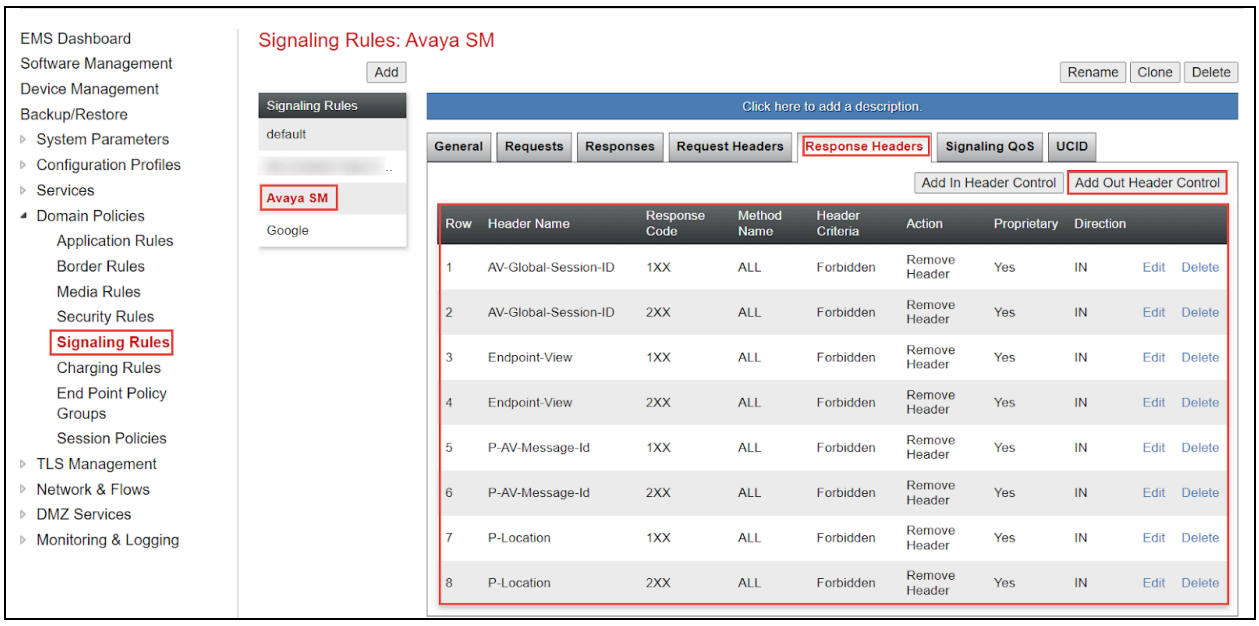

Figure 43: Signaling Rules for Avaya Aura SM (Cont.)

7.4.11 End Point Policy Groups

End Point Policy Group for **Avaya Aura SM**

- A new End Point Policy Group is created for Avaya Aura Session Manager.
- Navigate: **Domain Policies > End Point Policy Groups**
- Select **default-low** under Policy Groups
- Click **Clone**
- Set Group Name: **Avaya SM**
- Click **Next**

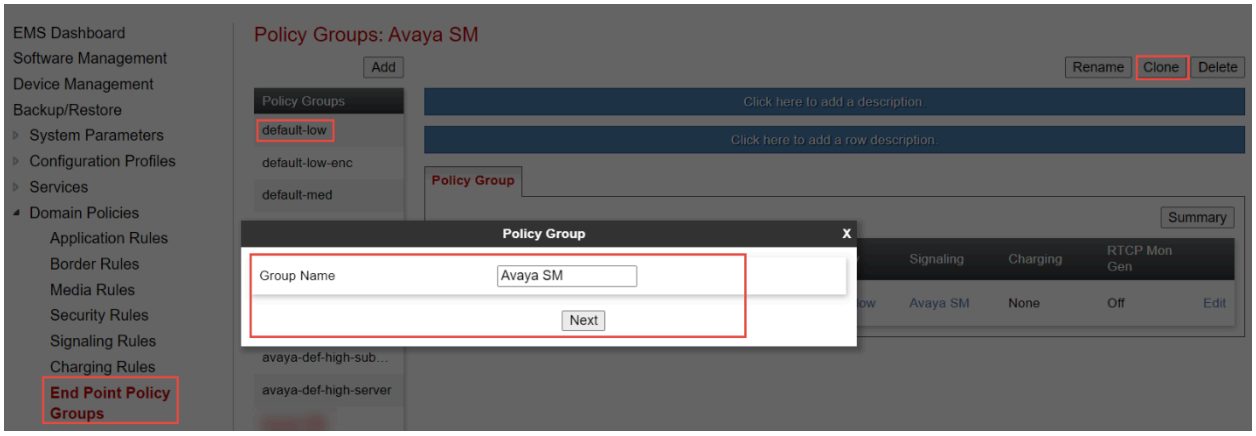

Figure 44: End Point Policy Group for Avaya Aura SM

- Select the newly created Group **Avaya SM**, Click **Edit**
- Set Signaling Rule: **Avaya SM**
- Click **Finish**

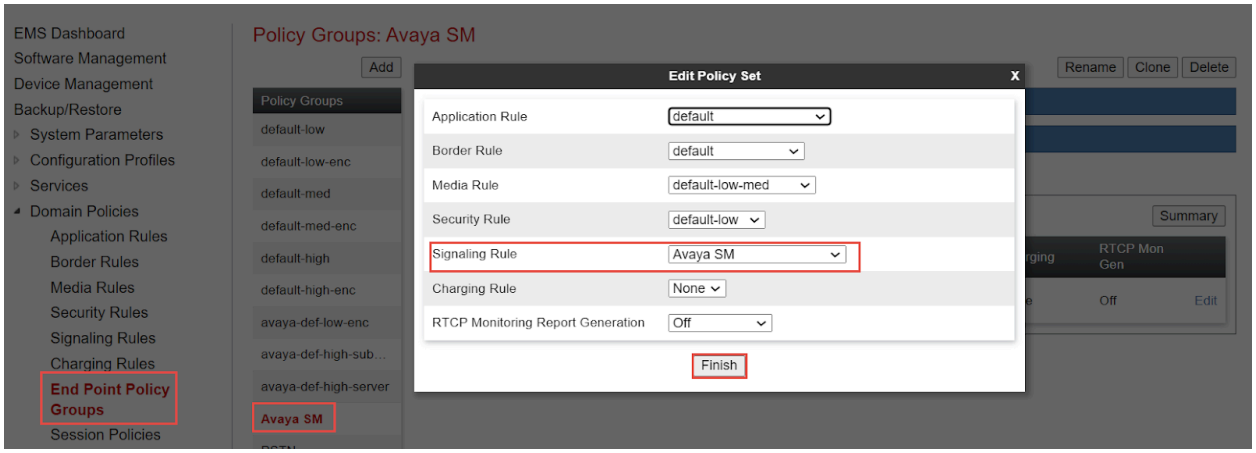

Figure 45: End Point Policy Group for Avaya Aura SM (Cont.)

End Point Policy Group for **Google CCAI**

- Repeat the same steps to create End Policy Group for **Google CCAI**

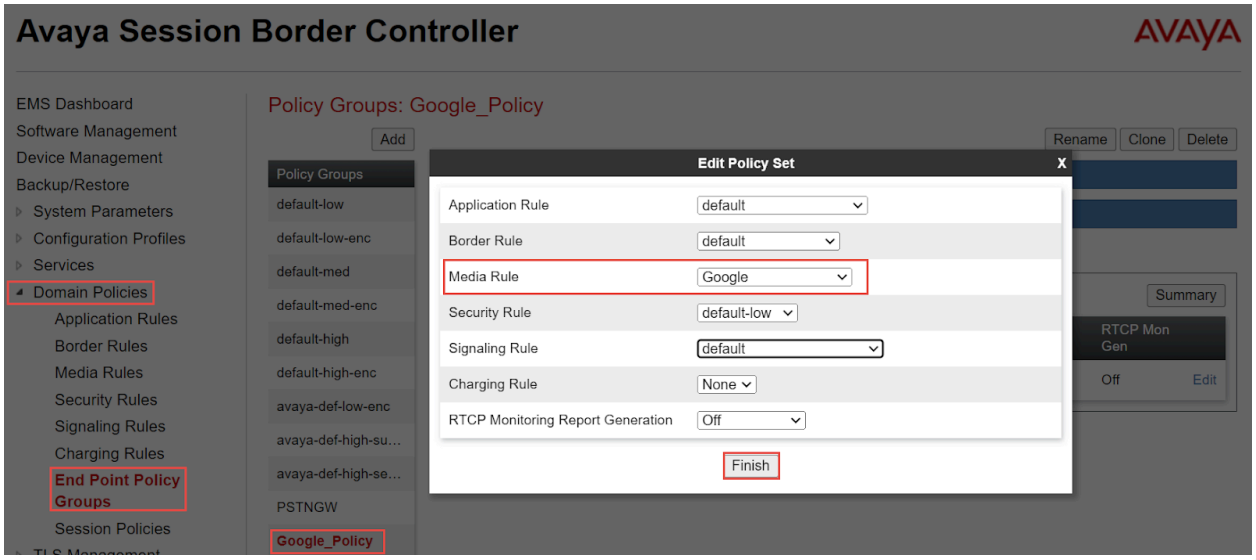

Figure 46: End Point Policy Group for Google CCAI

End Point Policy Group for **PSTN Gateway**

- Repeat the same steps to create End Policy Group for **PSTNGW**

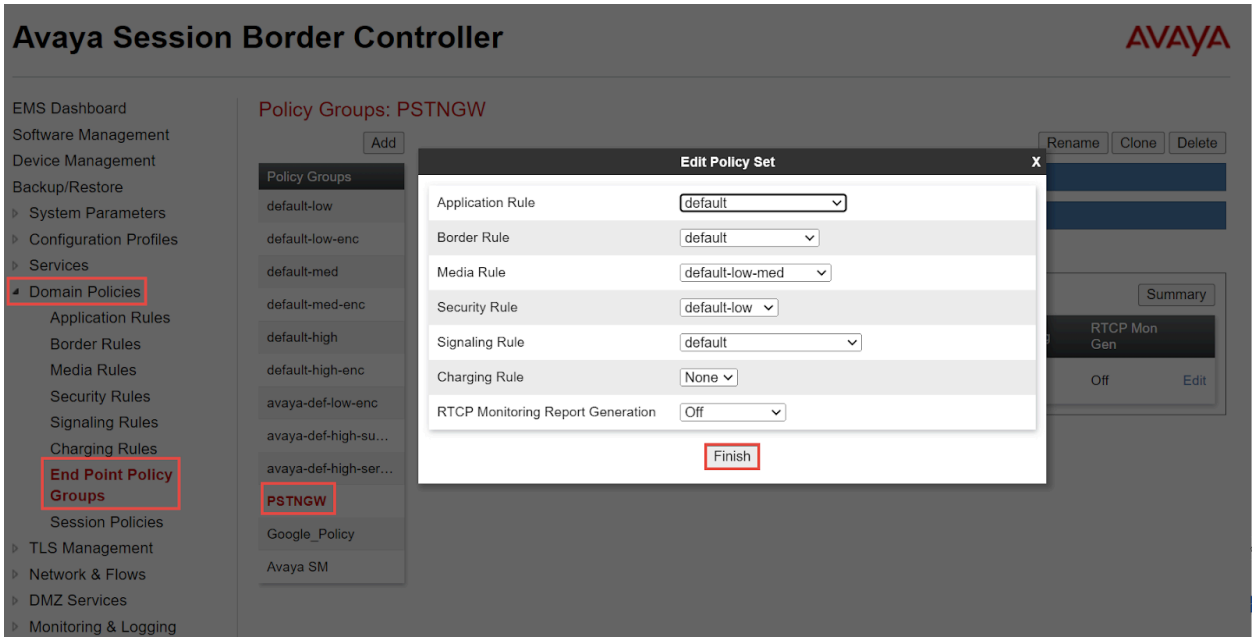

Figure 47: End Point Policy Group for PSTN Gateway

7.4.12 Media Interface

- Navigate: **Network & Flows > Media Interface**. Click **Add**
- Set Name: **AvayaSM10.2** is given here
- Set IP Address: Select LAN_PBX from the drop down and the IP address populates automatically. The IP address for Interface facing Avaya Aura SM is **10.70.X.X**
- Set Port Range: **35000-40000**
- Click **Finish**

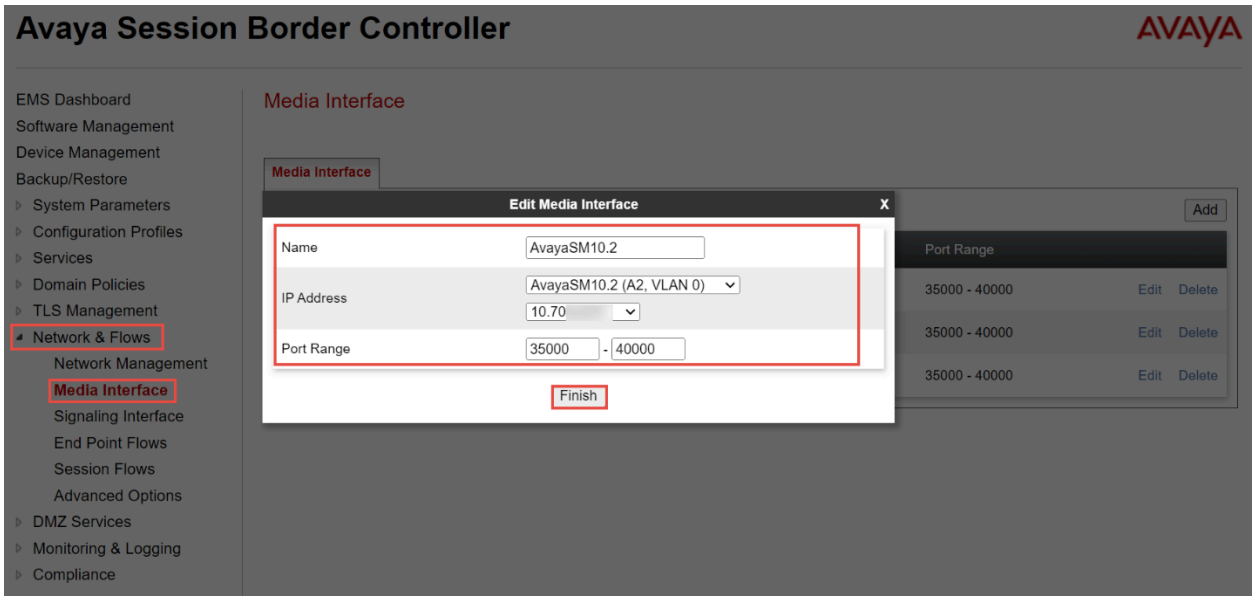

Figure 48: Media Interface Facing Avaya Aura SM

- Repeat the same steps to create a Media Interface facing **Google CCAI**

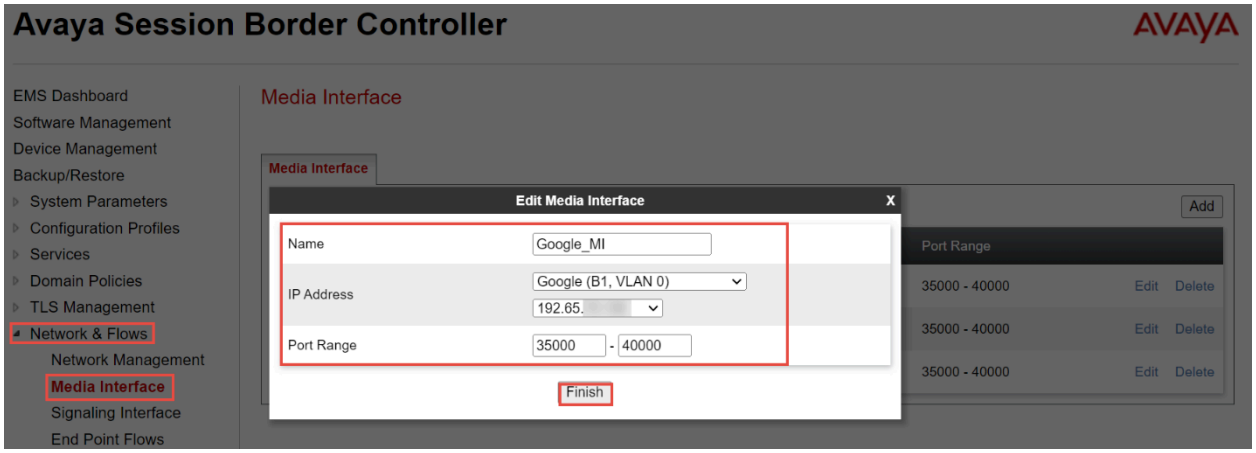

Figure 49: Media Interface Facing Google CCAI

- Repeat the same steps to create a Media Interface facing **PSTN Gateway**

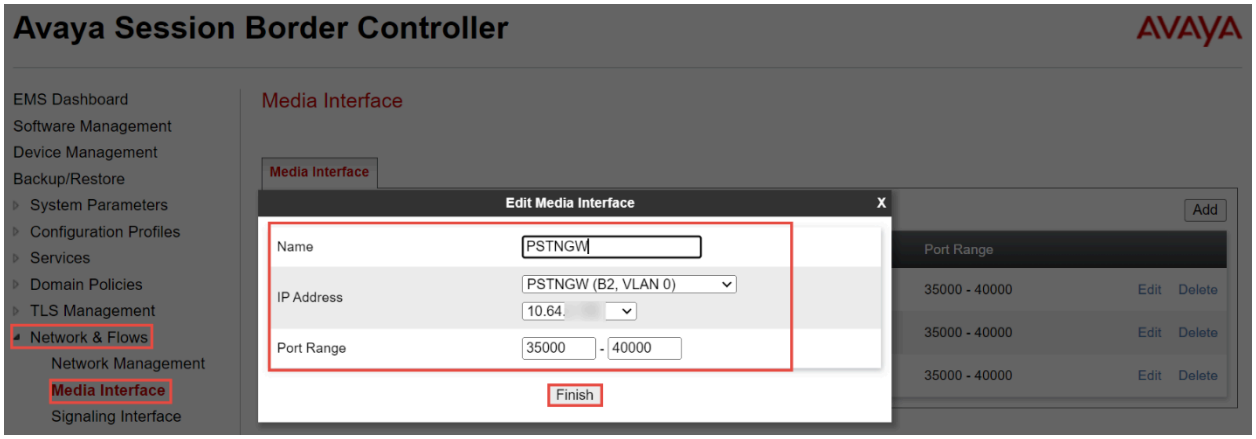

Figure 50: Media Interface Facing PSTN Gateway

7.4.13 Network Management

Network Management for **Avaya Aura SM**

- Navigate: **Network & Flows > Network Management**. Click **Add**, new Add Network Interface window appears
- Set Name: **AvayaSM10.2** is given for the network facing **Avaya Aura SM**
- Set **default Gateway IP Address**
- Set **Network Prefix or Subnet Mask**
- Set **Interface**
- Set **IP Address** facing Avaya Aura SM
- Click **Finish**

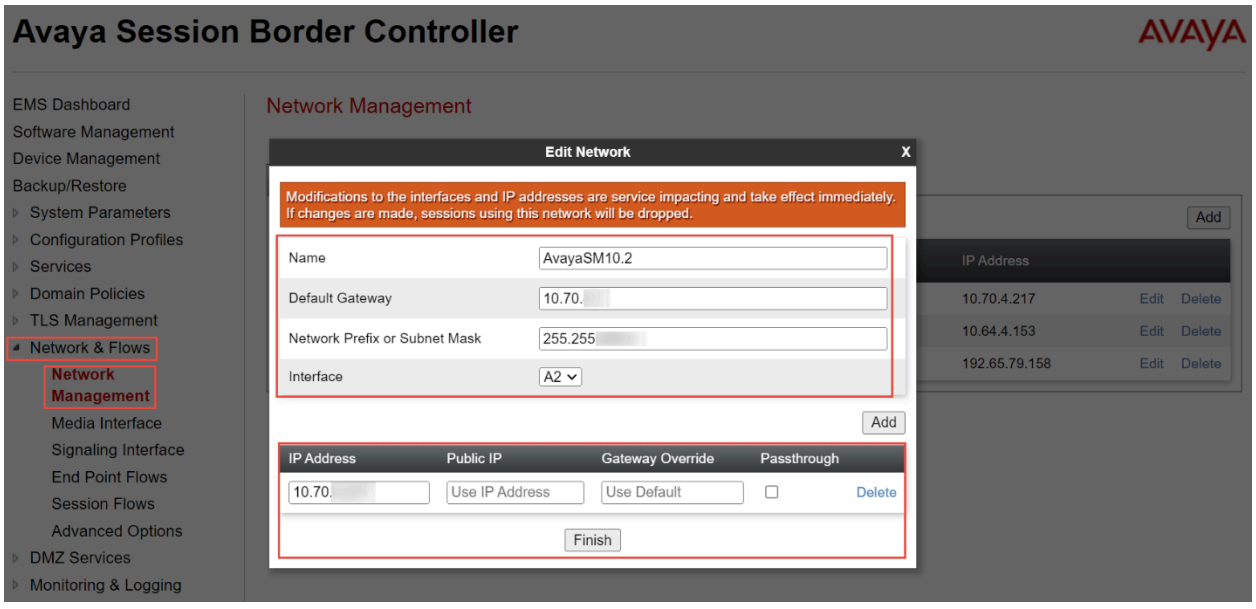

Figure 51: Network Management Facing Avaya Aura SM

Network Interface for **Google CCAI**

- Repeat the same steps to create the Signaling Interface facing **Google CCAI**.

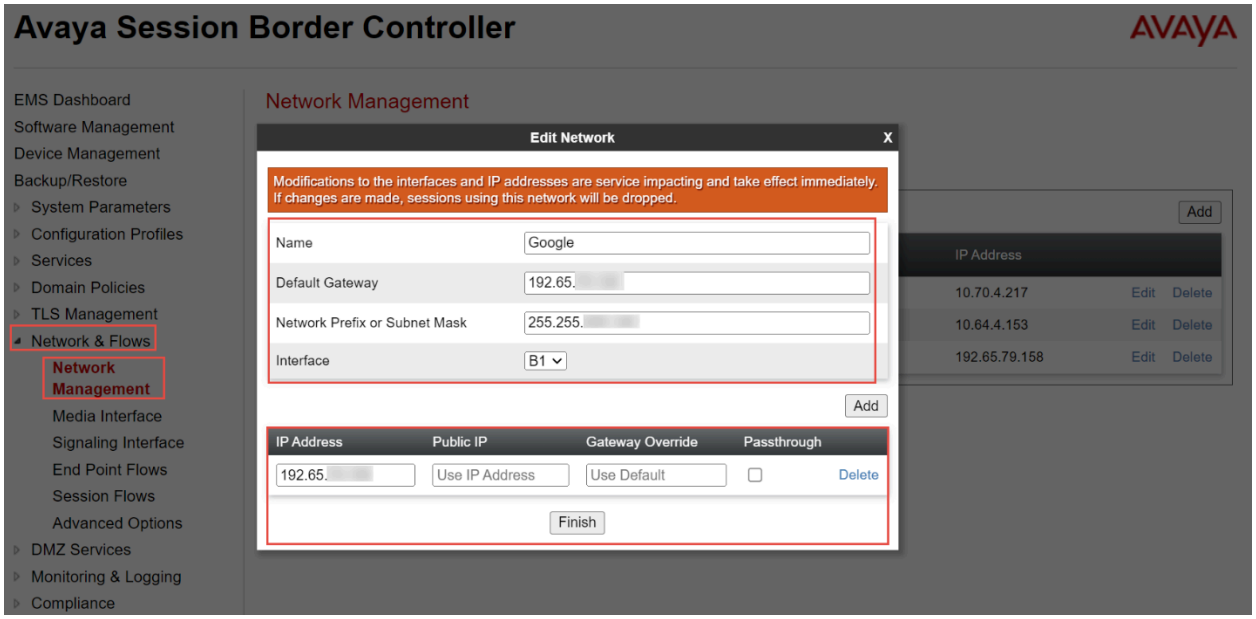

Figure 52: Network Management Facing Google CCAI

Network Interface for **PSTN Gateway**

- Repeat the same steps to create the Signaling Interface facing PSTN Gateway.

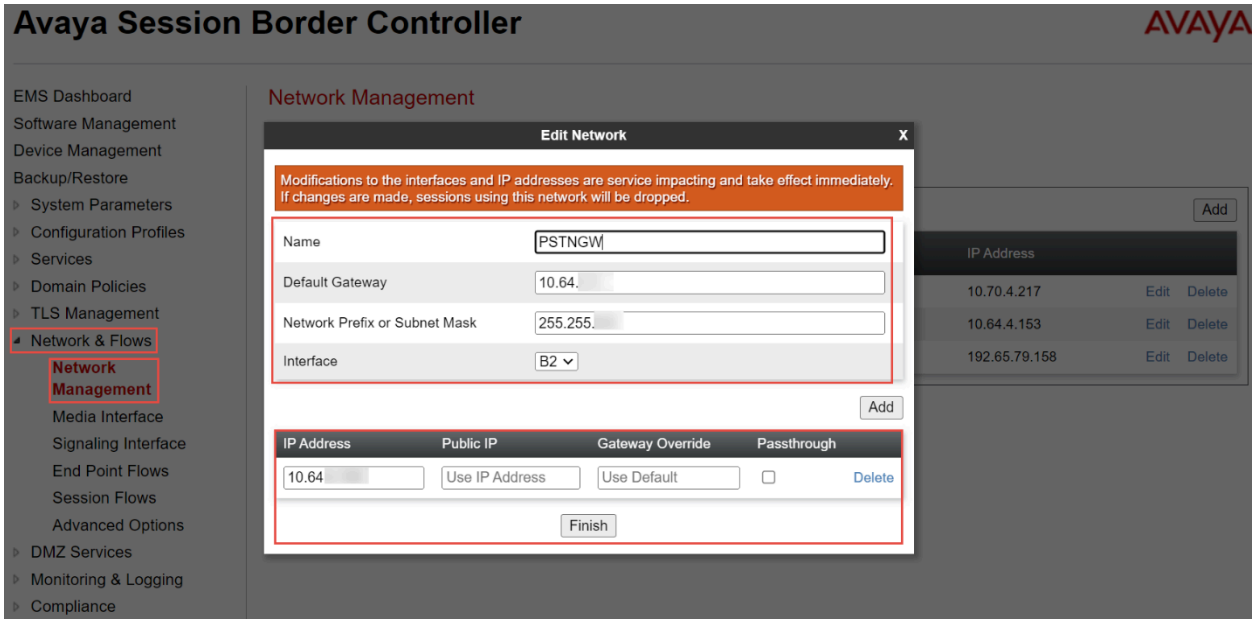

Figure 53: Network Management Facing PSTN Gateway

7.4.14 Signaling Interface

Signaling Interface for **Avaya Aura SM**

- Navigate to: **Network & Flows > Signaling Interface**. Click **Add**, new Add Signaling Interface window appears
- Set Name: **AvayaSM10.2** is given for the interface facing **Avaya Aura SM**
- Set IP Address: Select LAN_PBX
- Set TCP Port: **5060**
- Click **Finish**

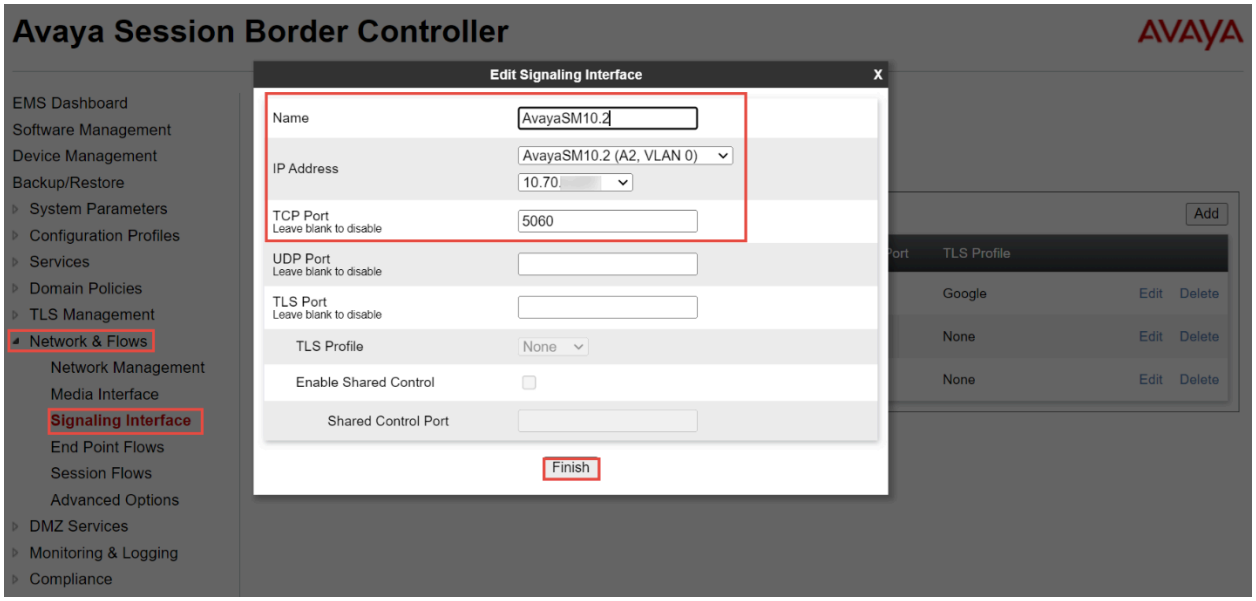

Figure 54: Signaling Interface Facing Avaya Aura SM

Signaling Interface for **Google CCAI**

- Repeat the same steps to create the Signaling Interface facing **Google CCAI**. TLS is used between Avaya SBC and Google CCAI.

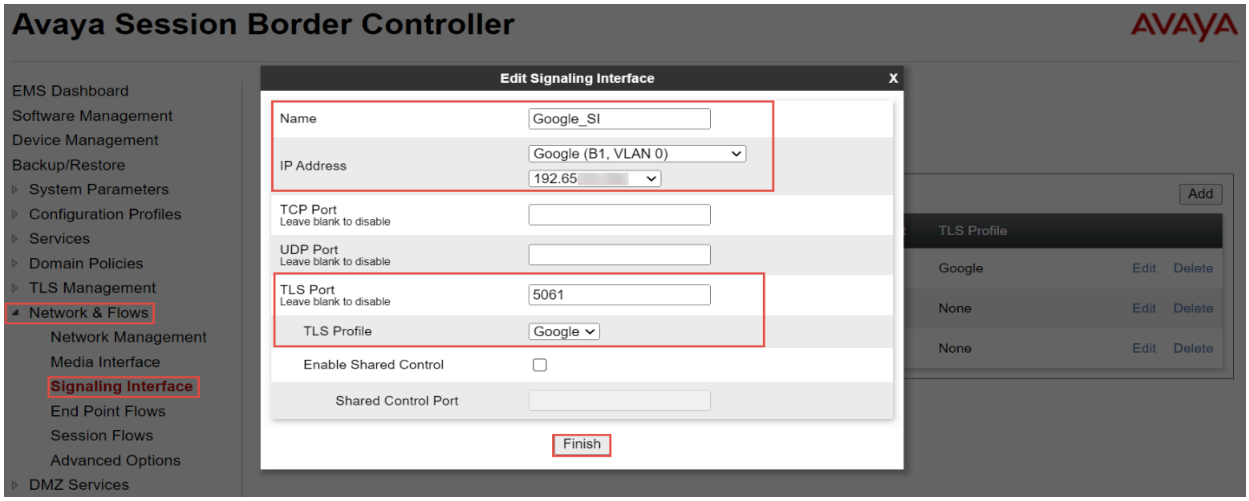

Figure 55: Signaling Interface Facing Google CCAI

Signaling Interface for **PSTN Gateway**

- Repeat the same steps to create the Signaling Interface facing PSTN Gateway. TCP is used between Avaya SBC and PSTN Gateway.

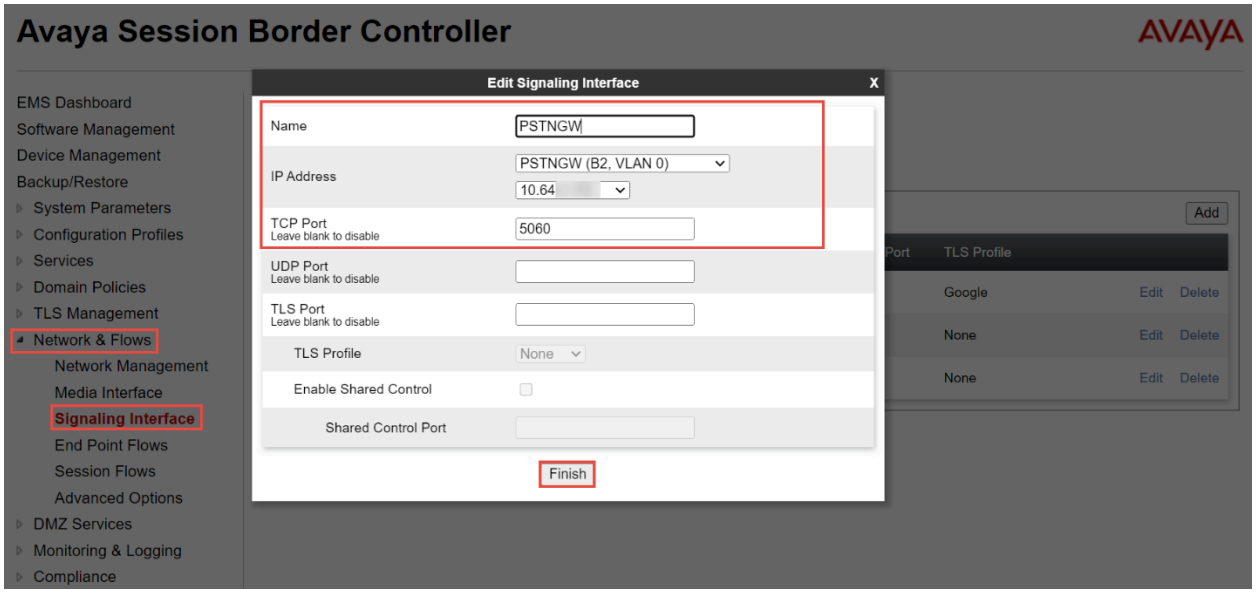

Figure 56: Signaling Interface Facing PSTN Gateway

7.4.15 End Point Flow

End Point Flow for **PSTN Gateway**

- Navigate: **Network & Flows > End Point Flows > Server Flows** Click **Add**
- Set SIP Server: **AvayaSM10.2**
- Select the required section: **Received Interface, Signaling Interface, Routing Profile and Topology Hiding Profile**

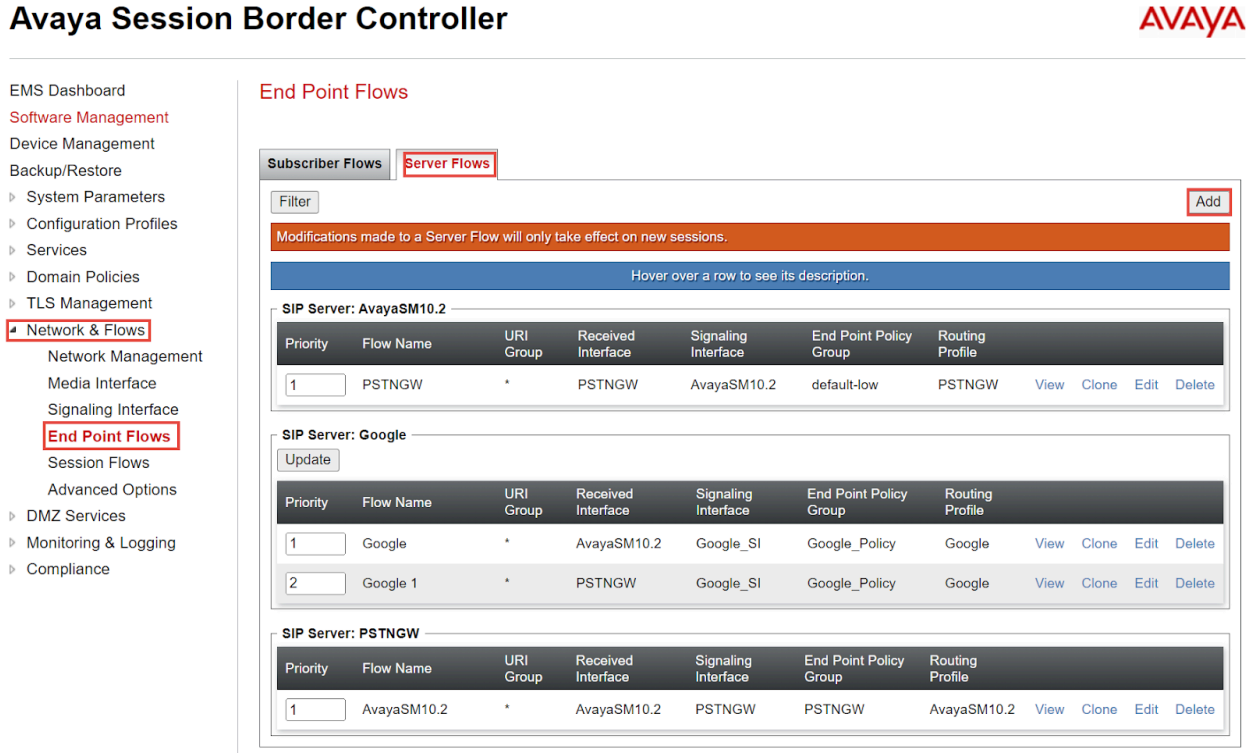

Figure 57: Server Flow for PSTN Gateway

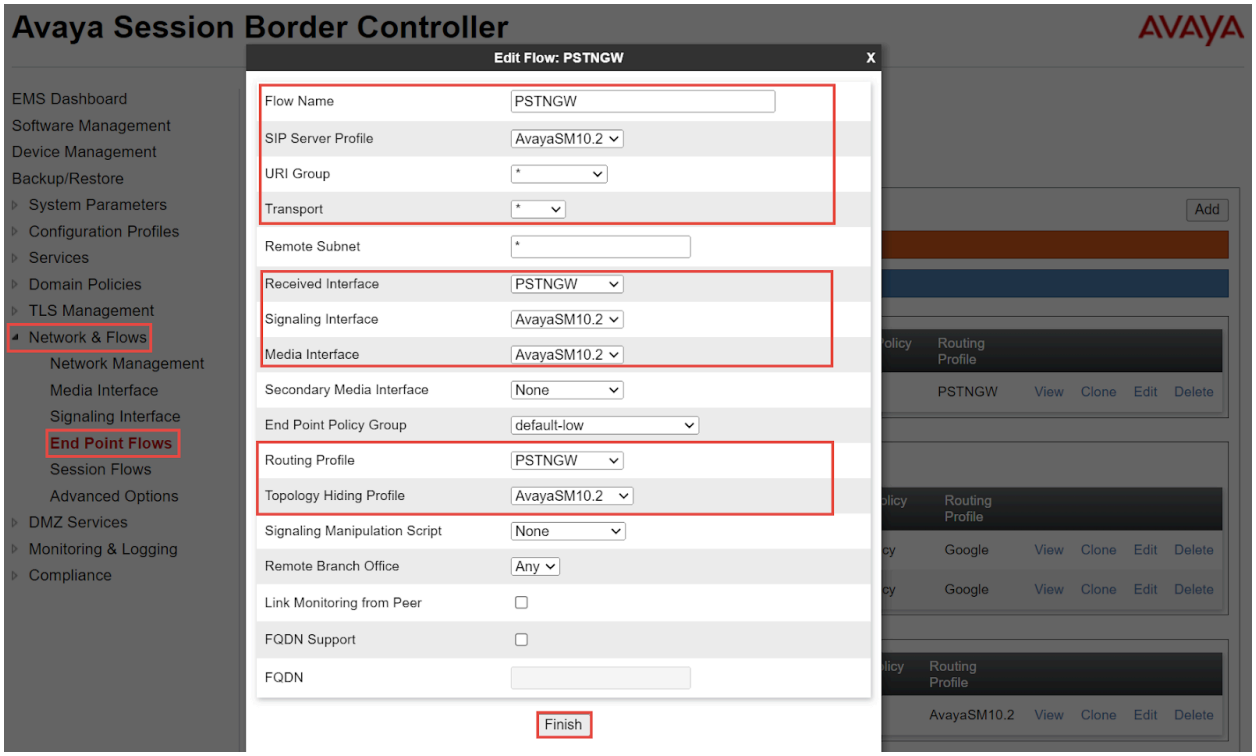

Figure 58: Server Flow for PSTN Gateway (Cont.)

End point flow for **Google CCAI**

- Navigate: **Network & Flows > End Point Flows > Server Flows** Click **Add**
- Set SIP Server: **Google**
- Select the required section: **Received Interface, Signaling Interface, Routing Profile, End Point Policy Group, Topology Hiding Profile and Signaling Manipulation script**

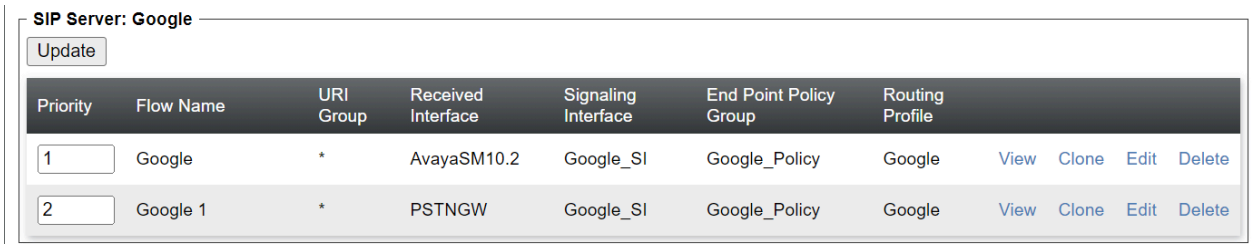

Figure 59: Server Flow for Google CCAI

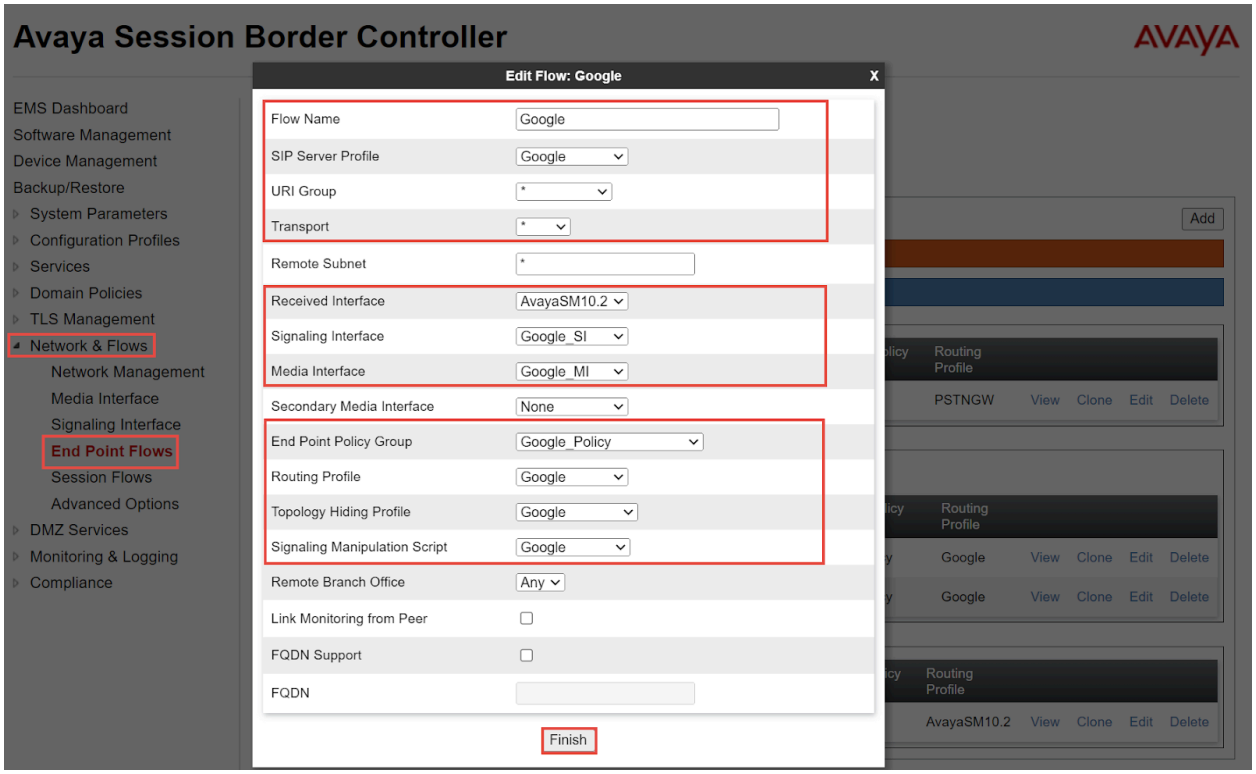

Figure 60: Server Flow for Google CCAI (Cont.)

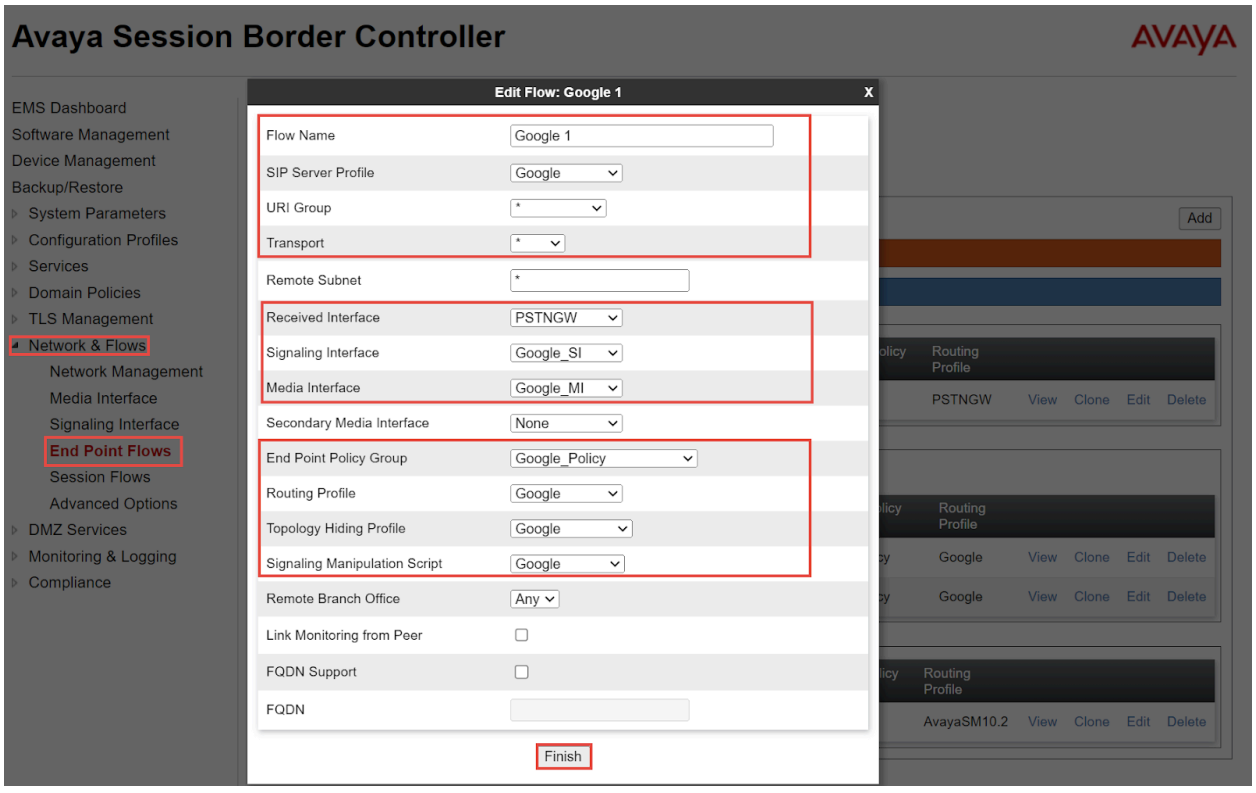

Figure 61: Server Flow for Google CCAI (Cont.)

End point flow for **Avaya Aura SM**

- Navigate: **Network & Flows > End Point Flows > Server Flows** Click **Add**
- Set SIP Server: **PSTNGW**
- Select the required section: **URI Group, Received Interface, Signaling Interface, Routing Profile, Topology Hiding Profile**

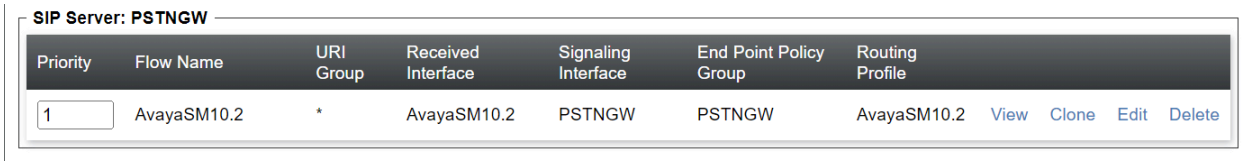

Figure 62: Server Flow for Avaya Aura SM

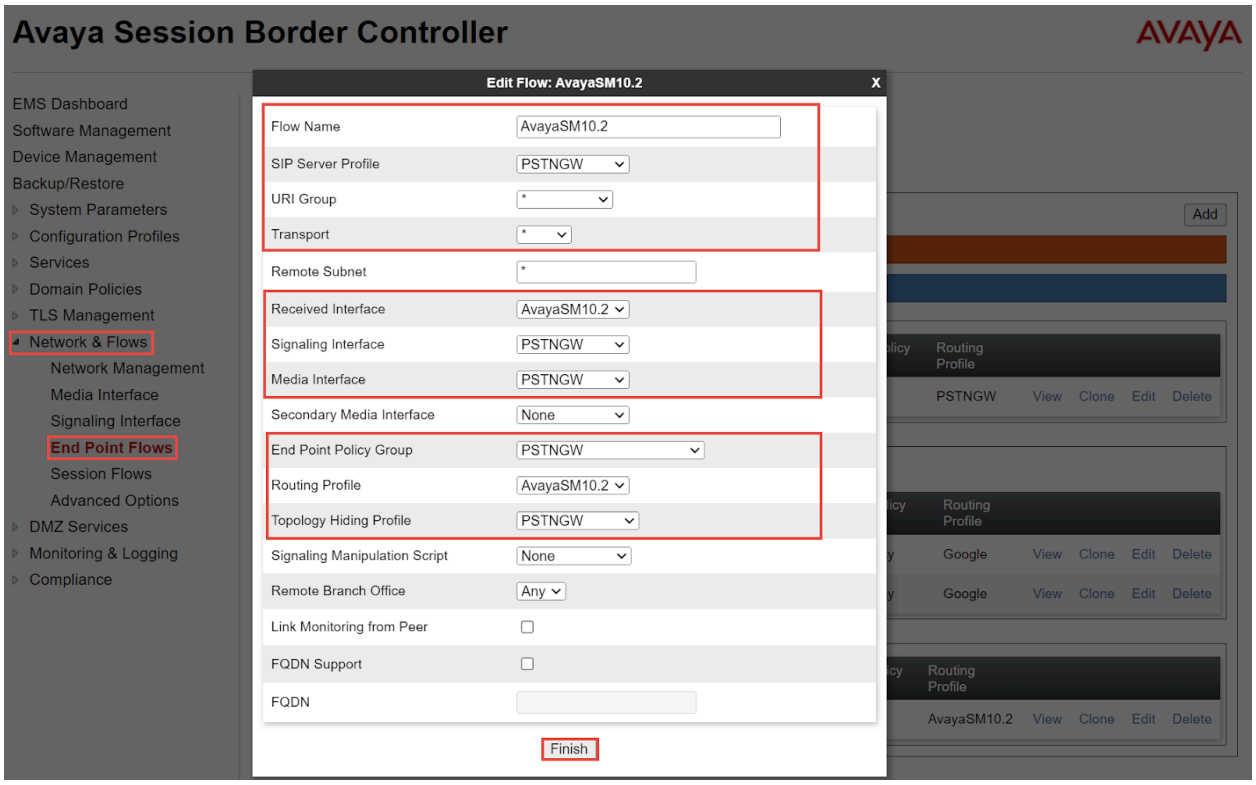

Figure 63: Server Flow for Avaya Aura SM (Cont.)

7.4.16 TLS Configuration

Creating SBC Certificate

- Navigate: **TLS management > Certificates**. Click **Generate CSR**

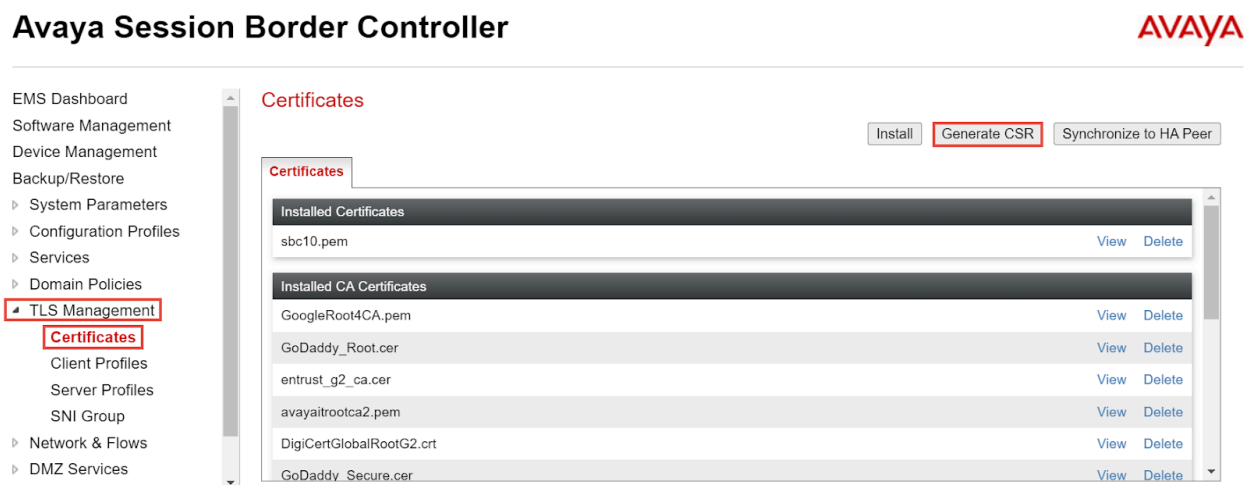

Figure 64: Generate CSR

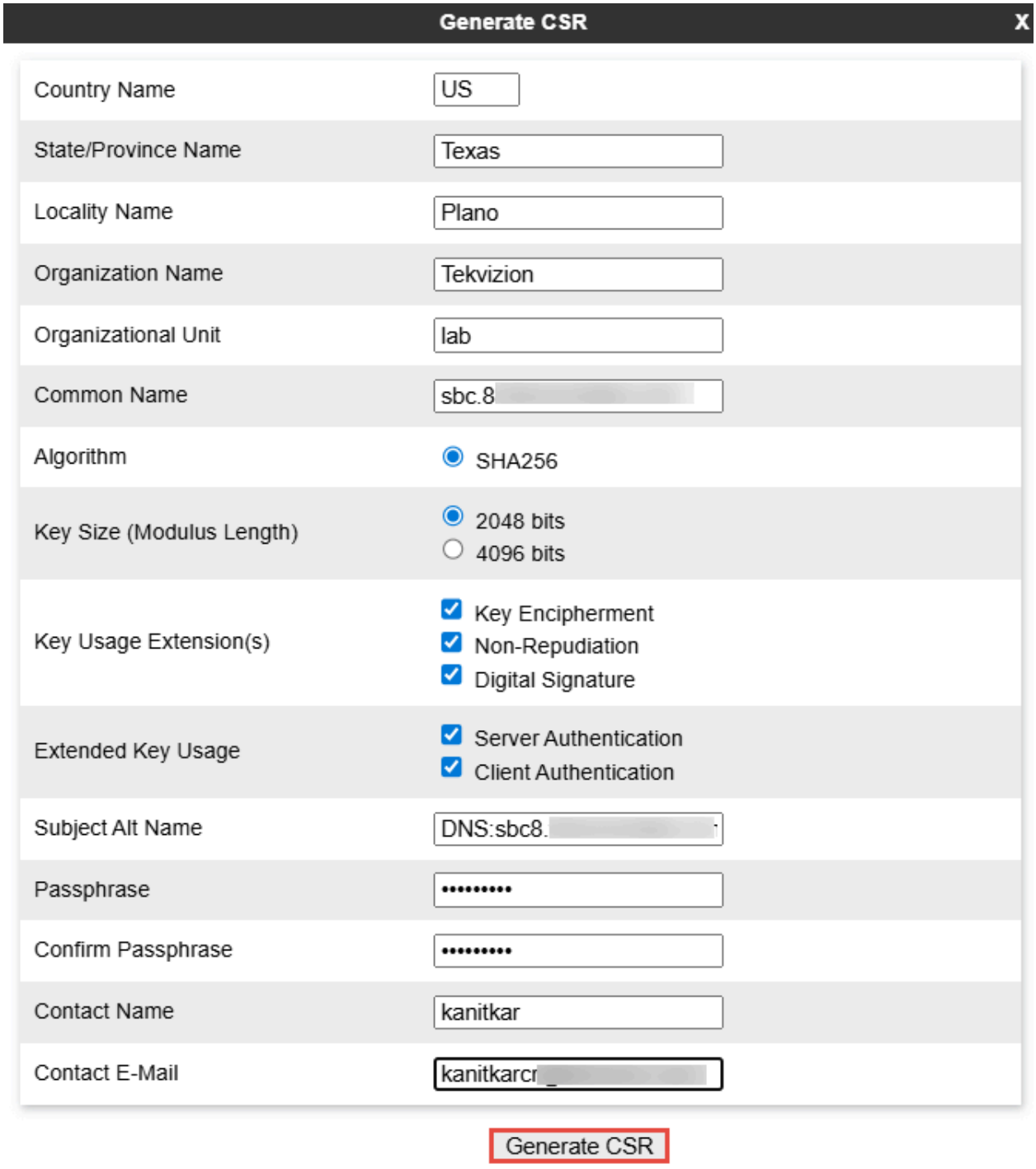

Figure 65: Generate CSR (Cont.)

- Navigate: **TLS management > Certificates**. Click **Install**
- Set Type: Select **CA Certificate**
- Set Name: **GoogleRoot1CA** (**GTS Root R1**)
- Set Allow weak Certificate/Key: Checked
- Set Certificate File: Click Choose File to select Google Root CA
- Click **Upload**
- Repeat the same steps to upload the GTS Root2.pem, GTS Root3.pem, GTS Root4.pem

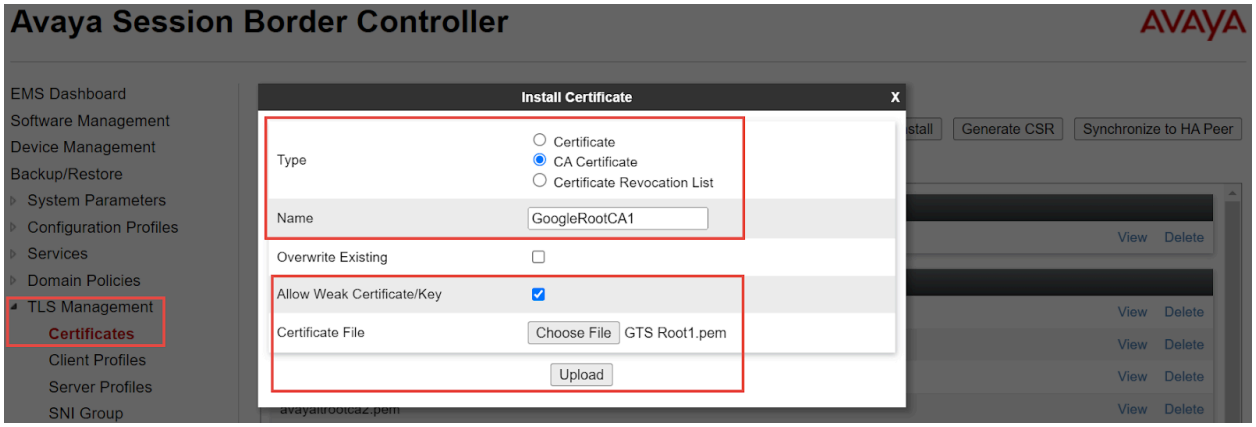

Figure 66: Upload Google Root CA

- Set Name: **GoDaddy_Root**
- Set Allow weak Certificate/Key: **Checked**
- Set Certificate File: Click Choose File to select **Go_Daddy_Root.cer**
- Click **Upload**

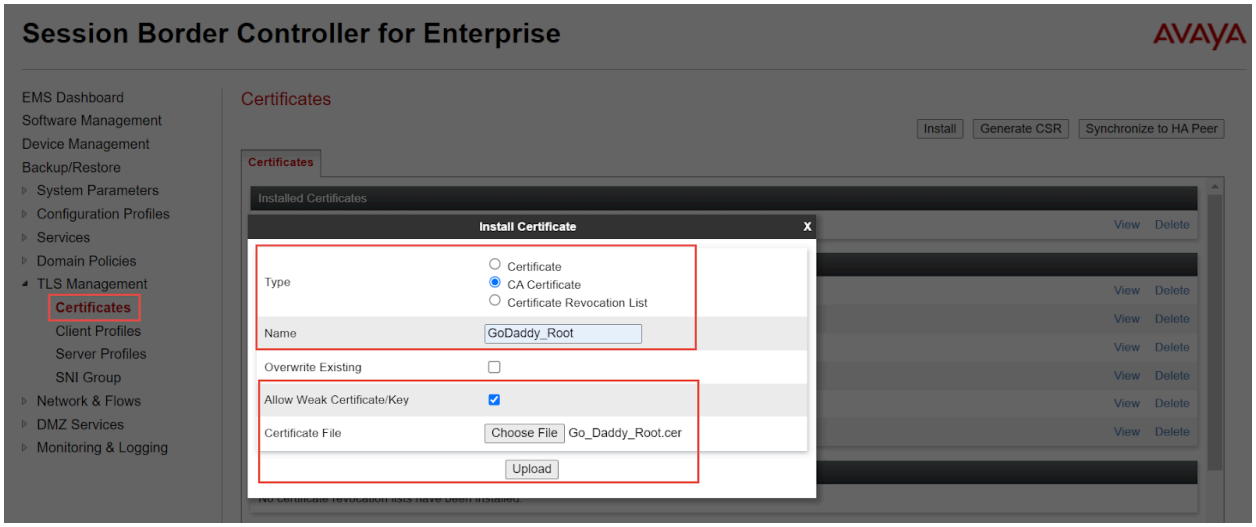

Figure 67: Upload GoDaddy Root CA

- Set Name: **Go_Daddy_Secure**
- Set Allow weak Certificate/Key: **Checked**
- Set Certificate File: Click Choose File to select **Go_Daddy_Secure.cer**
- Click **Upload**

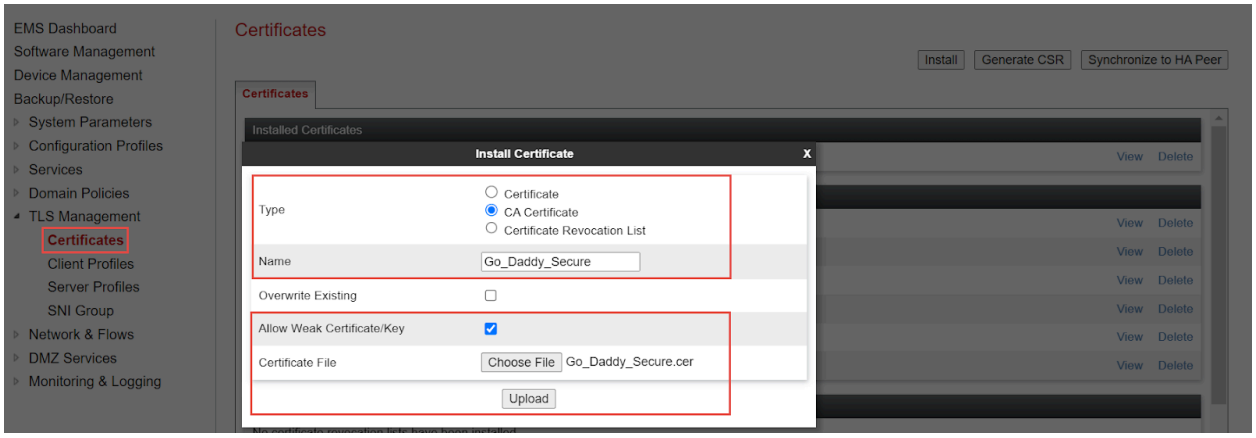

Figure 68: Upload GoDaddy Secure CA

- Navigate: **TLS management > Certificates**. Click **Install**
- Set Type: Select **Certificate**
- Set Name: **sbc8**
- Set Allow weak Certificate/Key: Checked
- Set Certificate File: Click Choose File to select **sbc10.pem**
- Select **Use Existing Key**
- Click **Upload**

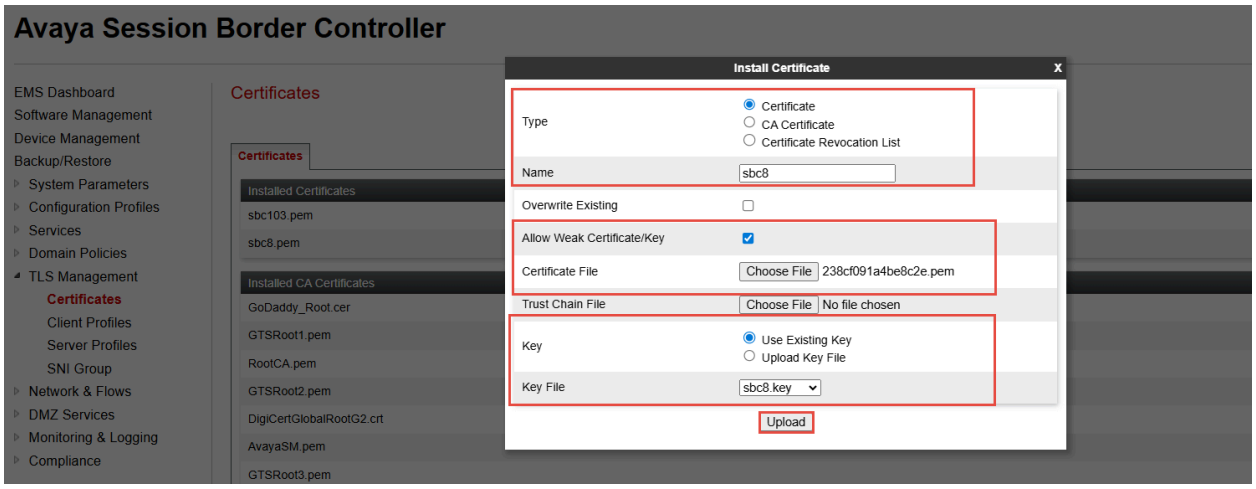

Figure 69: Upload SBC Certificate

Client Profile for **Google CCAI**

- Navigate: **TLS management > Client Profiles**. Click **Add**
- Set Profile Name: **Google** is given for interface facing Google
- Set Certificate: select server certificate **sbc8.pem** for Avaya SBC interface facing Google
- Set Peer Certificate Authorities: Select **GoogleRoot1CA.pem**, **GoogleRoot2CA.pem**, **GoogleRoot3CA.pem**, **GoogleRoot4CA.pem** which is uploaded in previous step
- Set Verification Depth: **5**

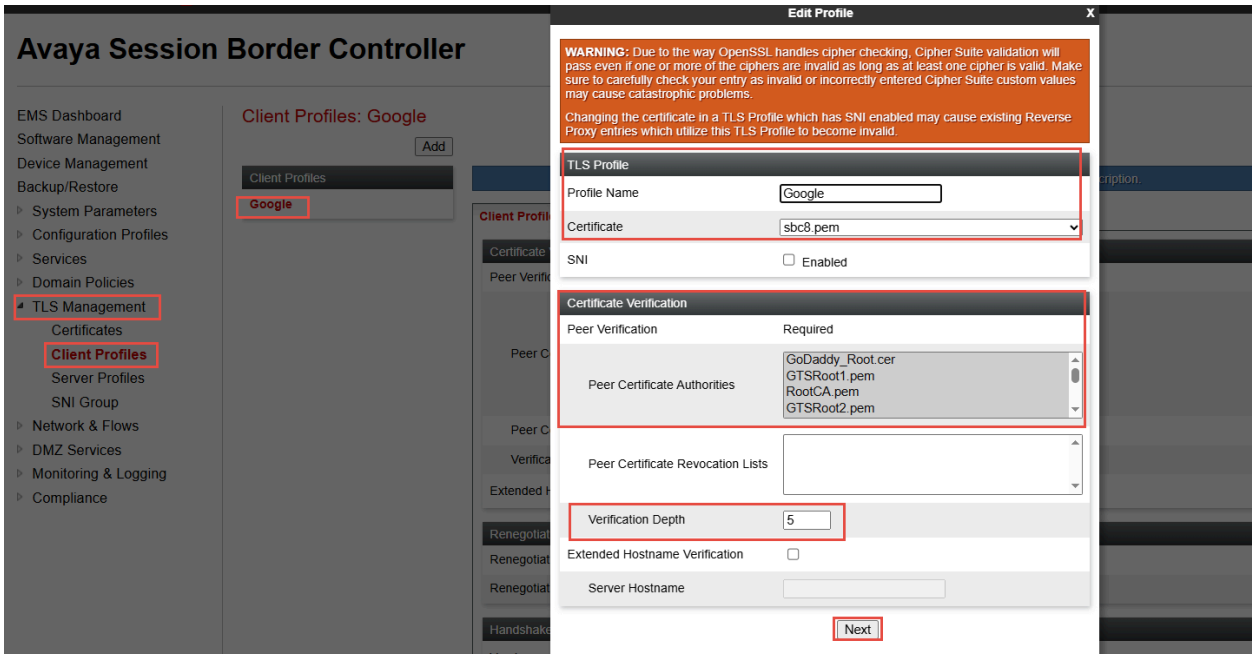

Figure 70: Client Profile Facing Google CCAI

- Set Version: Select **TLS 1.2** versions

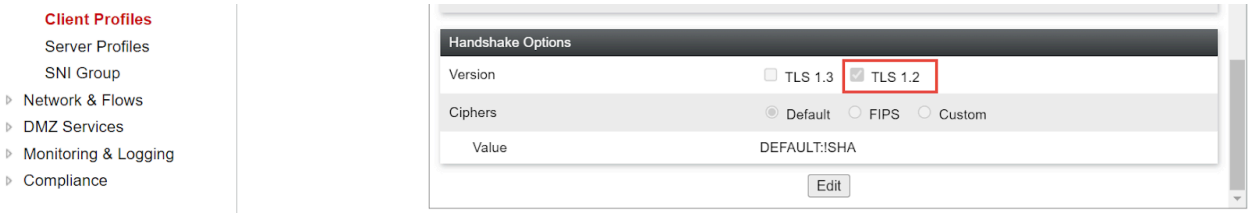

Figure 71: Client Profile Facing Google CCAI (Cont.)

Server Profile for **Google CCAI**

- Navigate: **TLS management > Server Profiles**. Click Add
- Set Profile Name: **Google** is given for interface facing Google
- Set Certificate: Select server certificate **sbc8.pem** for Avaya SBCE interface facing Google
- Set Version: Select **TLS 1.2** versions

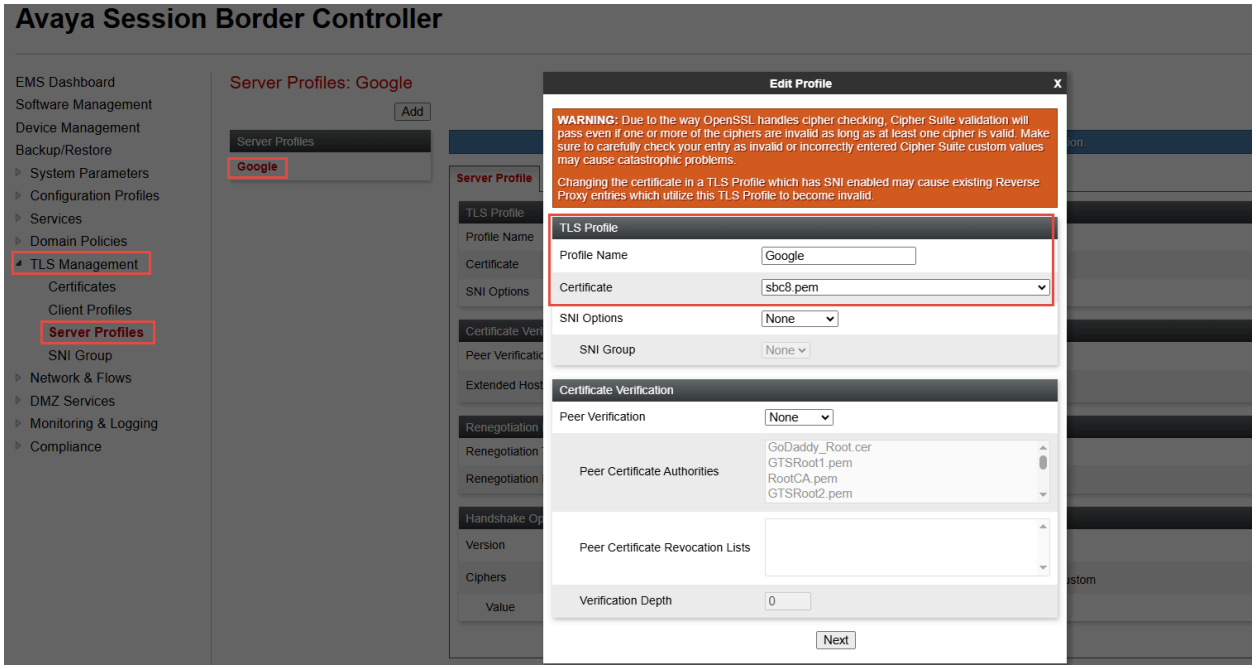

Figure 72: Client Profile Facing Google CCAI (Cont.)

Edit **SIP Server**

- Navigate: **Services > SIP Servers**
- Select Server Profile **Google**
- Under **General** tab, Click **Edit**
- Set Transport: Select **TLS** from Dropdown
- Set Port: **5672**
- Set TLS Client Profile: Select Client Profile **Google**
- Click **Finish**

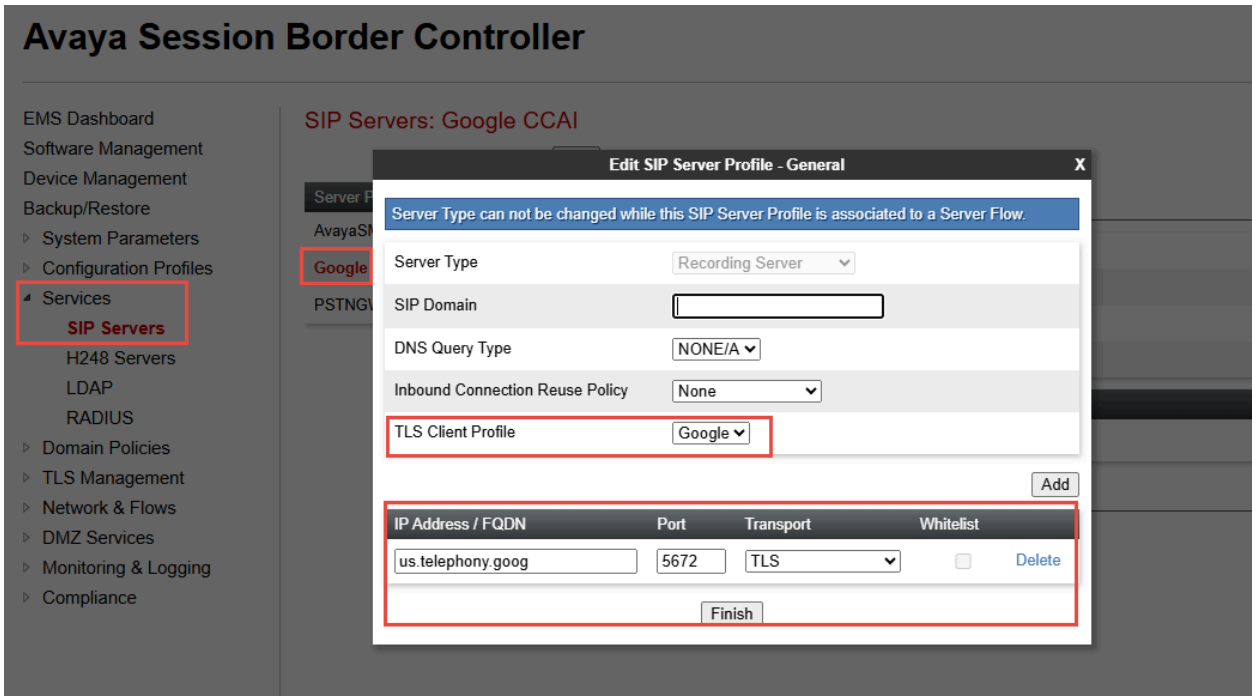

Figure 73: SIP Server Profile – Google CCAI

Configure SRTP

- Navigate: **Domain Policies > Media Rules**
- Select Media Rule default-low-med Click **Clone**
- Set Clone Name: **Google_MR**
- Click **Next**

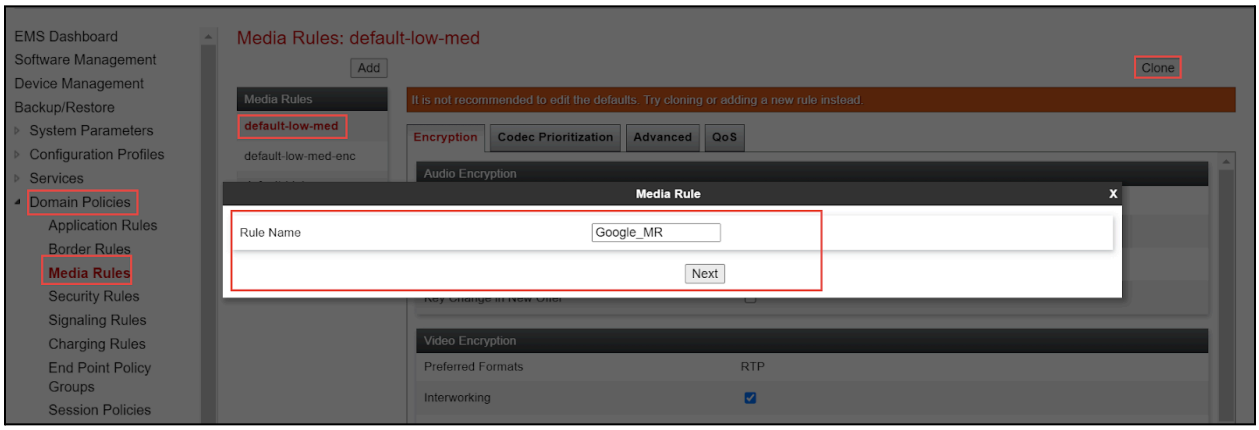

Figure 74: Media Rule – Google CCAI

- Select newly created Media Rule **Google**
- Set Preferred Format **SRTP_AES_CM_128_HMAC_SHA1_80**
- Set Encrypted RTCP: **checked**

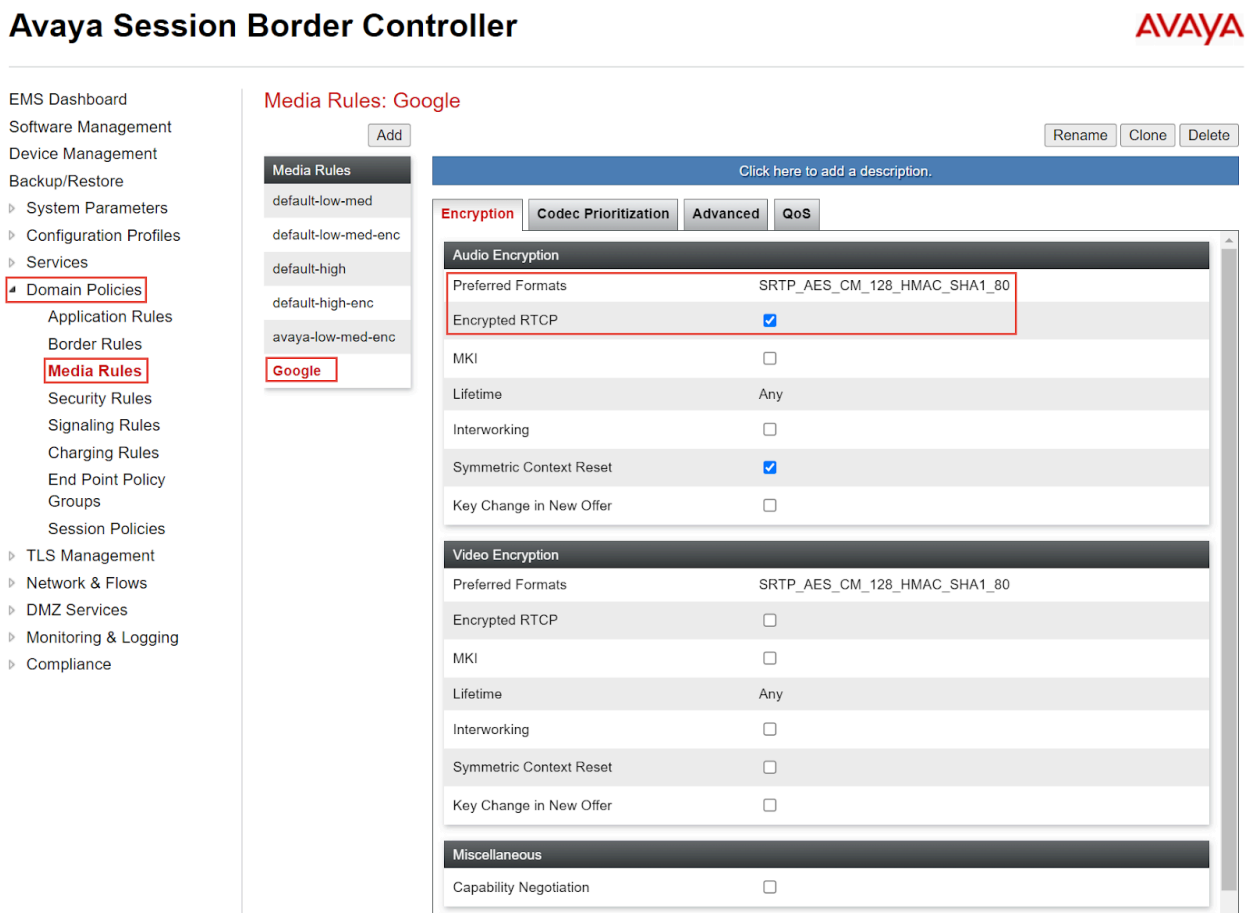

Figure 75:Media Rule– Google CCAI (Cont.)

Edit **End Point Policy Groups**

- Navigate to: **Domain Policies > End Point Policy Groups**
- Select **Google** under Policy Groups
- Click **Edit**

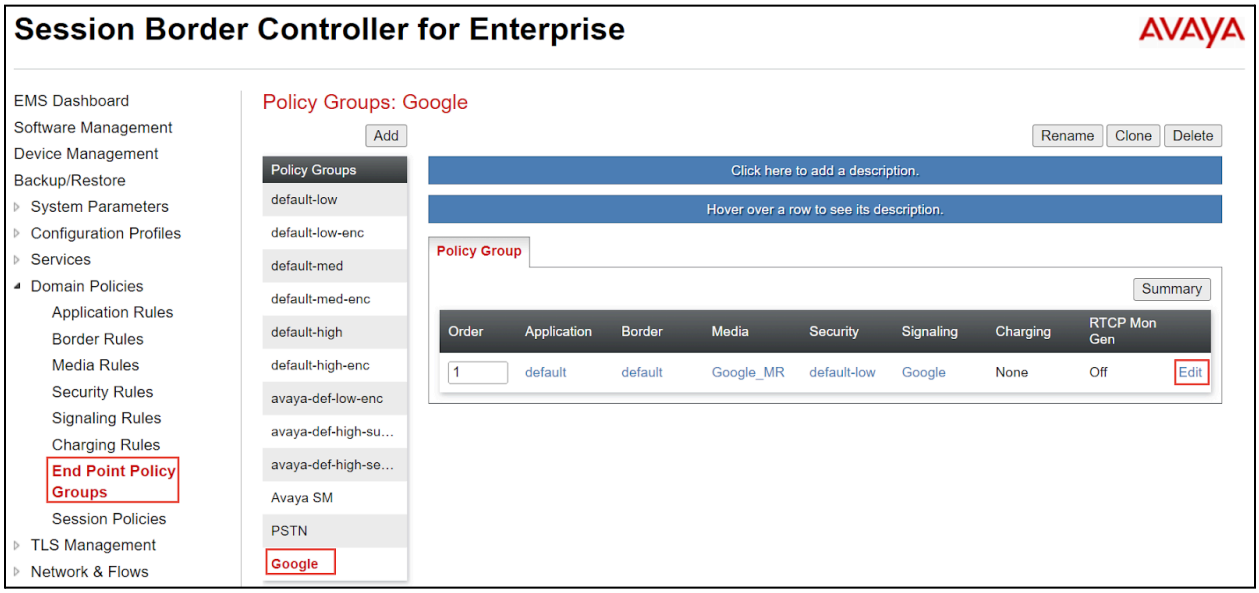

Figure 76:End Point Policy Group – Google CCAI

- Set **Media Rule**: Select **Google**
- Click **Finish**

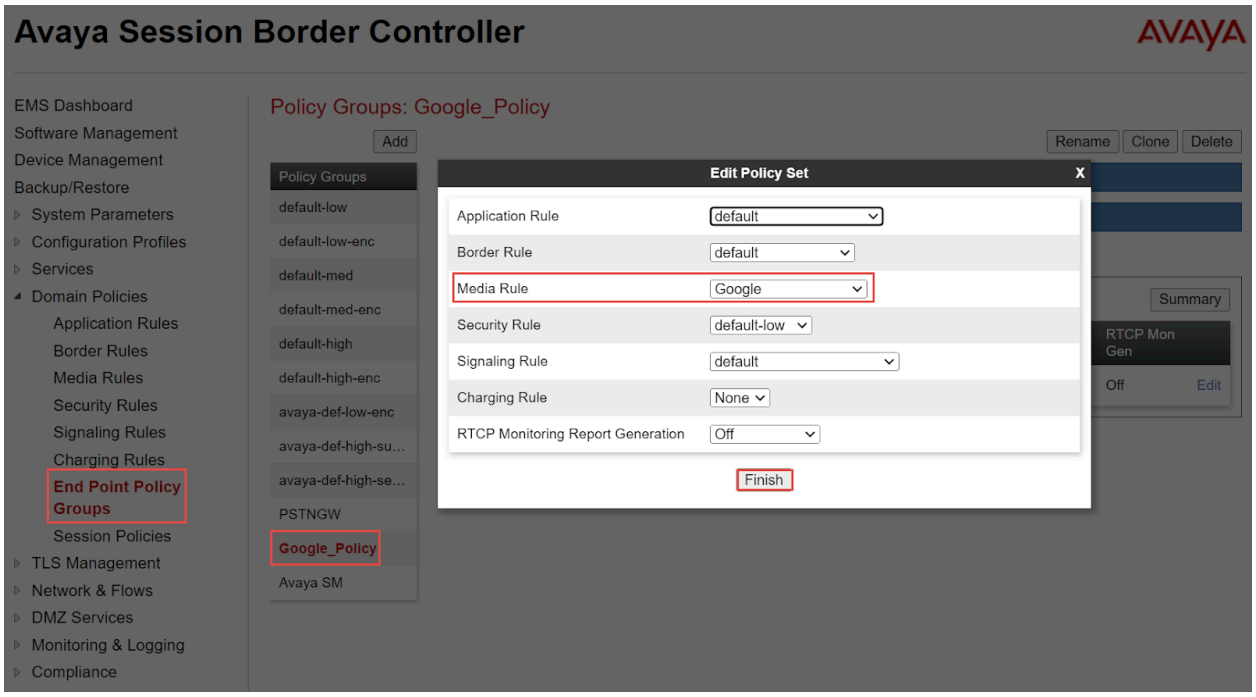

Figure 77:End Point Policy Group – Google CCAI (Cont.)

Edit **Signaling Interface**

- Navigate: **Network & Flows > Signaling Interface**
- Select interface **Google_SI**
- Click **Edit**

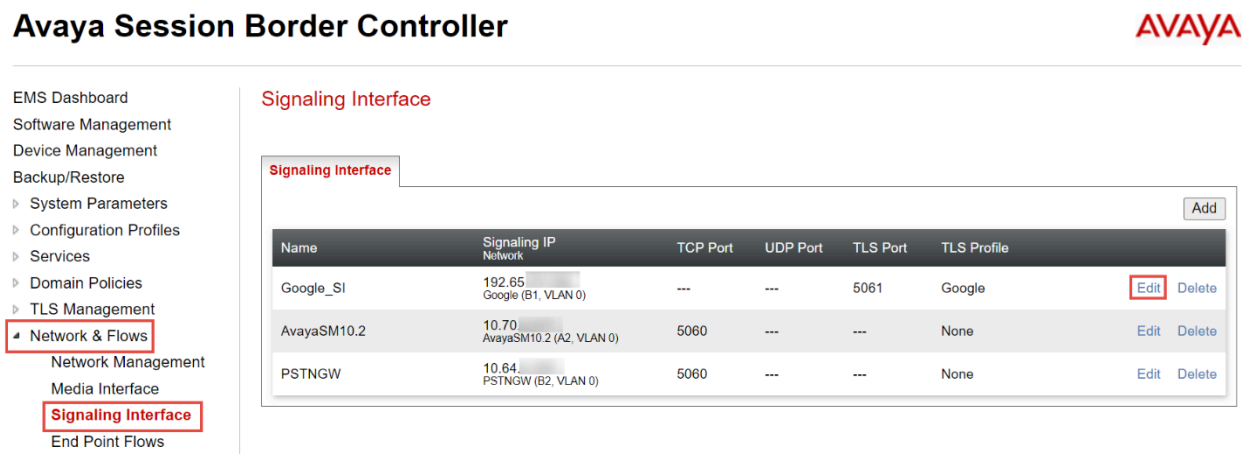

Figure 78:Signaling Interface – Google CCAI

- Set TLS Port: **5061**
- Set TLS Profile: Select **Google** from the drop-down menu
- Click **Finish**

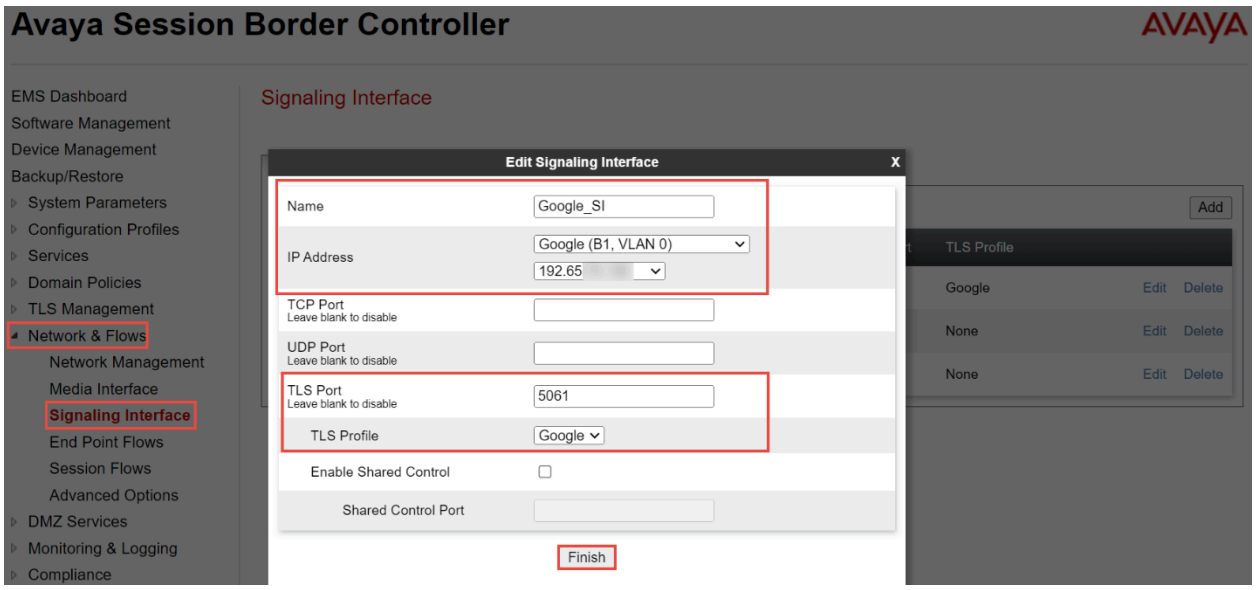

Figure 79:Signaling Interface – Google CCAI (Cont.)

8 Summary of Tests and Results

ID	Title	Description	Expected Results	Status (Passed or Failed etc)	Observations
SBC Configuration Verification					
1	SBC Configuration Verification	TLS connection setup. SBC initiates TLS connection with CCAI	Successful 4way handshake with Google CCAI. Validate the right certificates are being negotiated. SBC should be loaded with GTSR1 cert for Google. SBC should also send the certificate chain when sending its cert.	PASSED	
2	SBC Configuration Verification	TCP Keep Alive. SBC will perform monitoring checks by attempting TCP Keep Alive to ensure Network Connectivity	Successful 3way handshake and thereafter termination	PASSED	TCP Keep-alive packets are sent to the SIPREC Trunk
3	SBC Configuration Verification	TCP link is persistent. Establish calls, send multiple calls that should all use the same TCP transport connection	Persistent TCP connection, we should establish a single connection and multiplex all calls over that connection.	PASSED	
4	SBC Configuration Verification	Session Timer support. SBC should be initiator for the Session Refresh timer using Update or Re-Invite	Every 900 secs the SBC should refresh the SIP session.	PASSED	Avaya SBCE does not send session refresh re-invite. So Google sends refresh

ID	Title	Description	Expected Results	Status (Passed or Failed etc)	Observations
					sessions every 60 minutes using RE-INVITE
5	SBC Configuration Verification	SIP Header Manipulation (call-info header)	Validate if the Google requested header manipulation is present in the SIP INVITE. Ensure every SDP media has a label.	PASSED	
6	SBC Configuration Verification	*SBCs may need further Header manipulations based on SIP stack constraints. Verify required manipulation are added in SBC to support Google CCAI Example: FROM, TO header manipulations HOST part change in headers etc..,	All signaling in e.164 format	PASSED	
7	SBC Configuration Verification	SDES for SRTP. Configure the SDES parameters for crypto negotiation for the BYOT trunk	Validate the crypto is successfully negotiated and media is encrypted. All SBCs should support SDES for media encryption.	PASSED	
8	SBC Configuration Verification	DTLS for Media Encryption. Configure the DTLS parameters for crypto negotiation	Validate the crypto is successfully negotiated and	NOT SUPPORTED	Avaya SBC does not support DTLS

ID	Title	Description	Expected Results	Status (Passed or Failed etc)	Observations
		for the BYOT trunk, certificate for DTLS must be self-signed by the SBC.	media is encrypted.		
Inbound					
9	SIP OPTIONS	SBC send SIP options every 60 seconds	Verify SBC sends SIP OPTIONS every 60 seconds and responds with 200 OK	PASSED	
10	Inbound	Inbound call: Calling Party disconnects the call. Inbound siprec call, ensure recording are present, disconnect call from calling party and confirm proper disconnect	Verify Call is established with audio and transcripts from both participants Verify call is disconnected properly	PASSED	
11	Inbound	Inbound call: Called Party disconnects the call. Inbound siprec call, ensure recording are present, disconnect call from called party and confirm proper disconnect	Verify Call is established with audio and transcripts from both participants Verify call is disconnected properly	PASSED	
12	Inbound	Long duration call-Outbound Call- 1 hour max. Long duration siprec call	Ensure siprec calls stay up for an hour, confirm transcripts are present for entire duration	PASSED	Avaya SBCE does not send session refresh re-invite. So Google sends session refresh every 60 minutes using RE-INVITE

ID	Title	Description	Expected Results	Status (Passed or Failed etc)	Observations
13	Inbound	Long duration hold and resume (wait until session audit\session refresh occurs from DUT). Long duration siprec call, have the call placed on hold by agent, have call resume. Have customer place on hold then have call resume.	Call is connected, we have two active streams, confirm once a stream goes on hold, we receive corresponding signaling events, and that we no longer record transcripts for the participant on hold.	PASSED	Avaya SBCE does not send session refresh re-invite. So Google sends session refresh every 60 minutes using RE-INVITE
14	Inbound	Handling Error codes 603 decline. User A Calls PSTN A PSTN A rejects the incoming call	Verify SBC handles Call rejected properly	PASSED	
15	Inbound	Inbound call hold scenarios. Call starts out inactive for both participants, session moves to active	Validate if media is present when expected, confirm signaling events modify sdp properly, once call is move to active validate media and transcripts	PASSED	
16	Inbound	Inbound call hold scenarios. call starts out as active for both participants, session move to inactive, and transitions back to active	Validate if media is present when expected, confirm signaling events modify sdp properly, once call is moved to active validate media and transcripts	PASSED	

ID	Title	Description	Expected Results	Status (Passed or Failed etc)	Observations
17	Inbound	Update. Validate that update sent prior to call establishment do not contain SDP	Validate that update prior to call establishment do not contain SDP as expected	PASSED	UPDATE is sent from the SBC
18	Inbound	Update. Validate that updates post call establishment contain SDP to modify session	If SBC uses update to modify session, ensure SDP is included	NOT SUPPORTED	
19	Inbound	re-invites. Ensure re-invites that modify session include SDP	Ensure re-invites that modify session include SDP	PASSED	Re-INVITE is sent to Google CCAI as part of session refresh, hold scenarios
20	Inbound	Codec negotiation. Ensure that g711 u-law is preferred codec	Ensure we can prioritize g711 as preferred codec, note where SBC configures preferred codec	PASSED	
21	Inbound	3 way conference. Determine requirements, record all leg.	Determine requirements, record all legs	PASSED	
22	Inbound	CCAI cloud project setup. Establish CCAI cloud project, provision the project with a GTP phone number for access (Create conversations/participants on the fly through SIP headers)	Verify project is setup, functional test to confirm you can connect to the GTP access phone number	PASSED	

ID	Title	Description	Expected Results	Status (Passed or Failed etc)	Observations
23	Inbound	CCAI cloud project setup. Establish CCAI cloud project, provision the project with a GTP phone number for access (Pre-creation of conversations/participants)	Verify project is setup, functional test to confirm you can connect to the GTP access phone number	NOT APPLICABLE	This test case is not applicable for call recording
24	Inbound	Consultative transfer. Consultative transfer from 1. PSTN > User1 > User2 2. PSTN > User1 > PSTN user2		PASSED	
25	Inbound	Blind transfer. Blind transfer from 1. PSTN > User1 > User2 2. PSTN > User1 > PSTN user2		PASSED	Avaya PBX does not support blind transfer. This test case performed by ringing transfer
26	Validate Provisioning of trunk using self service	Validate Provisioning of trunk using self service	Use documentation to build trunk using self-service model	PASSED	
27	Inbound	Inbound call hold scenarios using a-law	Validate if media is present when expected, confirm Signaling events modify sdp properly, once call is move to hold active validate media and transcripts	PASSED	

ID	Title	Description	Expected Results	Status (Passed or Failed etc)	Observations
28	Inbound	Inbound call: Called Party disconnects the call. using a a-law codec	"Verify Call is established with audio and transcripts from both participants Verify call is disconnected properly Validate media stays in region"	PASSED	
29	Inbound	Long duration call-Outbound Call- 1 hour max using a-law codec	Ensure siprec calls stay up for an hour, confirm transcripts are present for entire duration.	PASSED	Avaya SBCE does not send session refresh re-invite. So Google sends session refresh every 60 minutes using RE-INVITE
30	Inbound	Inbound call: Configure trunk in non default region,	Verify Call is established with audio and transcripts from both participants Verify call is disconnected properly Validate media stays in region	PASSED	Testing conducted in the US region
31	Outbound	Participant Labels test	Configure call info header to specify roles, ensure the media streams align, Frist media stream HUMAN_AGENT role and Second is END_USER.	PASSED	When the roles are set to "HUMAN AGENT" and "END USER," (Call-Info<http://dialogflow.googleapis.com/v2beta1/projects/ccai-389811/conversation

ID	Title	Description	Expected Results	Status (Passed or Failed etc)	Observations
					s/Sr_17602595302582025718 16?roles=HUM AN_AGENT,EN D_USER>;purp ose=Goog-Co ntactCenter-C onversation) the transcript shows the first media stream with the participation role as "HUMAN AGENT," followed by "END USER."
32	Inbound	DTLS test		Not supporte d	
33	Inbound	Conference TEST	Determine requirements, record all legs	PASSED	
34	Inbound	Validate Call recording	Verify call recording is recorded throughout the call	PASSED	