Ribbon SBC Core SWe R10.1 Interop with NICE Engage 6.15 : Interoperability Guide

Table of Contents

Interoperable Vendors

Copyright

Document Overview

This document outlines the configuration best practices for theRibbon SBC SWe Core& PSX when deployed with NICE Recording Server.

About Ribbon SBC SWe Core :

The SBC SWe Core addresses the next-generation needs of SIP communications by delivering embedded media transcoding, robust security and advanced call routing in a high-performance, small form-factor device enabling service providers and enterprises to quickly and securely enhance their network by implementing services like SIP Trunking, secure Unified Communications and Voice over IP (VoIP).

The SBC SWe Coreprovides a reliable, scalable platform for IP interconnect to deliver security, session control, bandwidth management, advanced media services and integrated billing/reporting tools in an SBC appliance. This versatile series of SBCs can be deployed as peering SBCs, access SBCs or enterprise SBCs (eSBCs). The SBC product family is tested for interoperability and performance against a variety of third-party products and call flow configurations in the customer networks.

About Ribbon PSX :

The Ribbon PSX provides centralized policy and call routing engine for both Ribbon distributed Call Processing Node (CPN) such as GSX/SBC and also third-party call processing nodes.When deployed in Service Provider network or Enterprises network, it interfaces with these call processing nodes while processing either TDM (SS7, PRA) or SIP calls.

About NICE SIP Recorder :

The NICE Engage Platform provides comprehensive Omnichannel interaction recording to help organizations provide customers a coherent experience by providing a single place to define and implement compliance and quality practices across all channels.

Scope/Non-Goals

This document provides configuration best practices for deploying Ribbon's SBC SWe Core for NICE SIP recording Interop. Note that these are configuration best practices and each customer may have unique needs and networks. Ribbon recommends that customers work with network design and deployment engineersto establish the network design which best meets their requirements.

It is not the goal of this guide to provide detailed configurations that meet the requirements of every customer. Usethis guide as a starting point, and build the SBC configurations in consultation with network design and deployment engineers.

Audience

This is a technical document intended for telecommunications engineers with the purpose of configuring the Ribbon SBC SWe Core & PSX .

To perform this interop,you need to:

- use the graphical user interface (GUI) or command line interface (CLI) of the Ribbon product,
- understand the basic concepts of TCP/UDP/TLS and IP/Routing, and
- have understanding of SIP/RTP/SRTP to complete the configuration and for troubleshooting.

Note

This configuration guide is offered as a convenience to Ribbon customers. The specifications and information regarding the product in this guide are subject to change without notice. All statements, information, and recommendations in this guide are believed to be accurate but are presented without warranty of any kind, express or implied, and are provided AS IS. Users must take full responsibility for the application of the specifications and information in this guide.

Prerequisites

The following aspects are required before proceeding with the interop:

- Ribbon SBC SWe Core
- Ribbon SBC SWe Core license
  - A valid license from Ribbon is required toenable functionality on RibbonSBCs. Each SBC license provides a base set of capabilities to allow enabling and adding of additional features and capacity, as required.
- TLS certificatesfor SBC SWe Core
  - Please refer toManaging Certificates
- Ribbon PSX
- NICE Engage setup

NICE VRSP server functions as a SIP Proxy to set up SIP sessions between the SBC and the NICE.VRSP internally communicates to NICE AIR server which acts as a recording server. In active standby mode we have two VRSP servers with one as Active and one as Standby. Throughout this document from SBC perspective , we will be mentioning VRSP server as SRS[Session Recording Server].

Product and Device Details

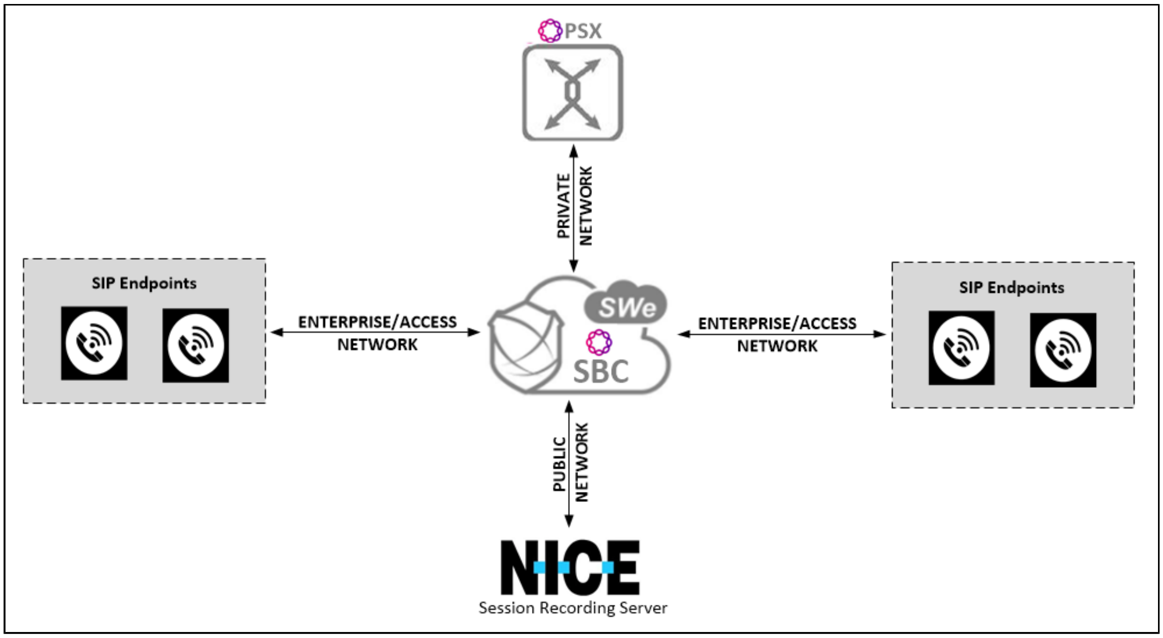
The configuration uses the following equipment and software:

Product	Equipment/Service	Software Version
Ribbon Communications	SBC SWe Core	V10.01.00-R000
	PSX	V14.1
Third-Party Equipment	NICE Recording Server	V6.15
Endpoints	PhonerLite	V2.96
	Zoiper5	V5.5.8
Administration and Debugging Tools	Wireshark	V3.0.1

Network Topology and E2E Flow Diagrams

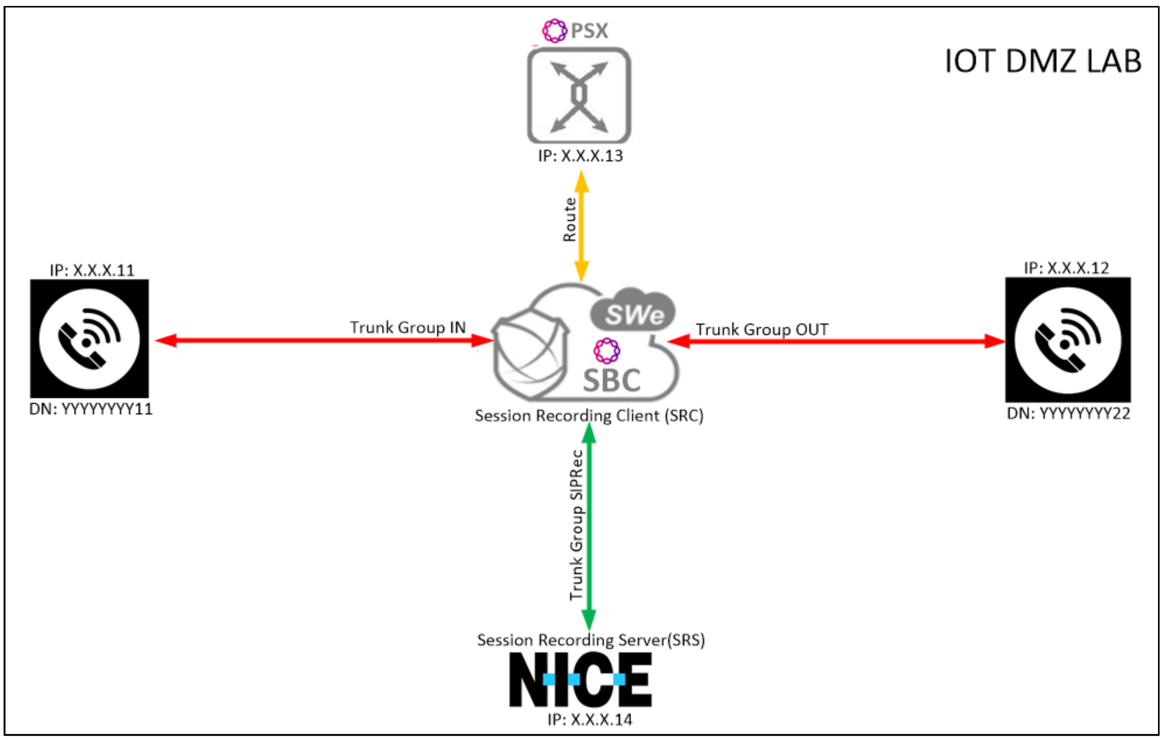
Deployment Topology

Figure 1:



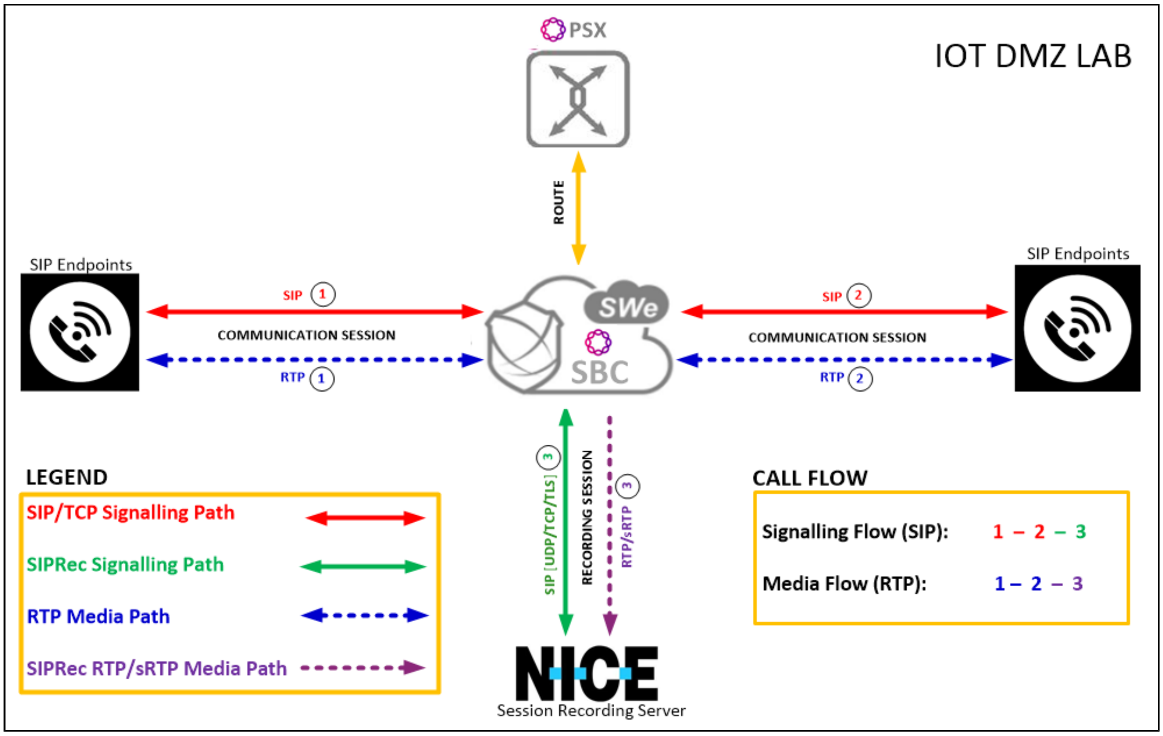
Interoperability Test Lab Topology

Figure 2:



Call Flow Diagram

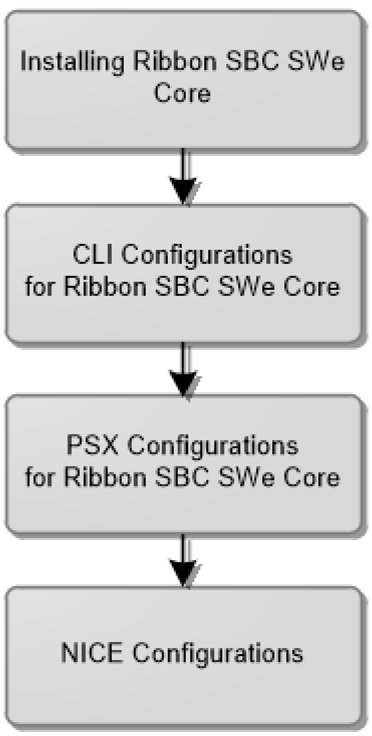
Figure 3:



Document Workflow

The sections in this document follow the sequence below. The reader is advised to complete each section for the successful configuration.

Figure 4:



Installing Ribbon SBC SWe Core

Ribbon SBC Standalone

To deploy Ribbon SBC SWe Core StandAlone instance, refer toSBC Core 10.1.x Documentation

Ribbon SBC High Availability

To deploy Ribbon SBC SWe Core in HA mode on different platforms, refer toSBC Core Software Installation and Upgrade Guide

During this interop, SBC SWe Core HA was installed on VMware platform by following the procedure described inInstalling SBC Application in High Availability Mode.

- After successful installation, ensure the time on both Active and Standby SBCs is in sync.
- NTP Sync verification:
  - Run the command 'timedatectl' to check if NTP is synchronized.
  - File /etc/ntp.conf should contain the IP of the NTP server that you have configured during installation

CLI Configurations forRibbonSBC SWe Core

Global Configuration

1.Configure IP Interface Group

An IP Interface Group is a named object containing one or more IP interfaces (IP addresses). The IP Interface Group is Address Context-specific (e. g. permanently bound to a particular Address Context), and is the primary tool to manage disjointed networks (separate networks that are not designed to communicate directly). An IP Interface Group is the local manifestation of a segregated network domain. The service section of an IP trunk group and a Signaling Port typically reference an IP Interface Group in order to restrict signaling and/or media activity to that IP Interface Group.

```
set addressContext default ipInterfaceGroup IG1 ipInterface IP1 ceName SBCSIPREC
set addressContext default ipInterfaceGroup IG1 ipInterface IP1 portName pkt0
set addressContext default ipInterfaceGroup IG1 ipInterface IP1 ipAddress <The primary IP address of the interface>
set addressContext default ipInterfaceGroup IG1 ipInterface IP1 prefix <The IP subnet prefix of this Interface>
set addressContext default ipInterfaceGroup IG1 ipInterface IP1 mode inService
set addressContext default ipInterfaceGroup IG1 ipInterface IP1 state enabled
set addressContext default ipInterfaceGroup IG2 ipInterface IP2 ceName SBCSIPREC
set addressContext default ipInterfaceGroup IG2 ipInterface IP2 portName pkt1
set addressContext default ipInterfaceGroup IG2 ipInterface IP2 ipAddress <The primary IP address of the interface>
set addressContext default ipInterfaceGroup IG2 ipInterface IP2 prefix <The IP subnet prefix of this Interface>
set addressContext default ipInterfaceGroup IG2 ipInterface IP2 mode inService
set addressContext default ipInterfaceGroup IG2 ipInterface IP2 state enabled
commit
```

2. Configure Static Route

IP Static Routeobject specifies the gateway to which you wish to direct traffic from your Packet, Management, or Link Interface. In effect, this object allows you to add, change, and delete gateways (next Hops) to these interfaces. Interface and static routes combine to form the IP routing table for your network.

An IP Static Route provides a route to each potential call destination IP address. The static route is used to add static IP routes for the IP interfaces. A static route indicates the next Hop gateway and IP interface to use for a particular peer network IP prefix.

```
set addressContext default staticRoute <destinationIpAddress> 0 <nextHopIPaddress> IG1 IP1 preference 100
set addressContext default staticRoute <destinationIpAddress> 0 <nextHopIPaddress> IG2 IP2 preference 100
commit
```

SBC Configuration for Endpoints

1. Create new Zone and configure sipSigPort

A Zoneis used to group a set of objects unique to a particular customer environment.

A SIP Signaling Port is a logical address permanently bound to a specific zone, and is used to send and receive SIP call signaling packets. A SIP Signaling Port is capable of multiple transports such as UDP, TCP, and TLS/TCP.

```
set addressContext default zone zone1 id 111
set addressContext default zone zone1 sipSigPort 1 ipInterfaceGroupName IG1
set addressContext default zone zone1 sipSigPort 1 ipAddressV4 <IPv4 address>
set addressContext default zone zone1 sipSigPort 1 portNumber <1-65535>
set addressContext default zone zone1 sipSigPort 1 mode inService
set addressContext default zone zone1 sipSigPort 1 state enabled
set addressContext default zone zone1 sipSigPort 1 transportProtocolsAllowed sip-udp,sip-tcp,sip-tls-tcp
set addressContext default zone zone2 id 222
set addressContext default zone zone2 sipSigPort 2 ipInterfaceGroupName IG2
set addressContext default zone zone2 sipSigPort 2 ipAddressV4 <IPv4 address>
set addressContext default zone zone2 sipSigPort 2 portNumber <1-65535>
set addressContext default zone zone2 sipSigPort 2 mode inService
set addressContext default zone zone2 sipSigPort 2 state enabled
set addressContext default zone zone2 sipSigPort 2 transportProtocolsAllowed sip-udp,sip-tcp,sip-tls-tcp
commit
```

2. Create basic Trunk Group Configurations

SIP Trunk Groupsare used to apply a wide-ranging set of call management functions to a group of peer devices (endpoints) within the network. SIP Trunk Groups are created within a specific address context and zone.

All SBCsignaling and routing (both Trunking and Access) are based upon Trunk Group configurations defined within zones. A zone can contain multiple Trunk Groups.

Please ensure to configure similar transport preferences in CLI and PSX Trunk Group configurations

```
set addressContext default zone zone1 sipTrunkGroup SIPREC_TG1 signaling transportPreference preference1 tcp
set addressContext default zone zone1 sipTrunkGroup SIPREC_TG1 media mediaIpInterfaceGroupName IG1
set addressContext default zone zone1 sipTrunkGroup SIPREC_TG1 ingressIpPrefix <IP address> <prefix>
set addressContext default zone zone1 sipTrunkGroup SIPREC_TG1 state enabled
set addressContext default zone zone1 sipTrunkGroup SIPREC_TG1 mode inService
set addressContext default zone zone2 sipTrunkGroup SIPREC_TG2 signaling transportPreference preference1 tcp
set addressContext default zone zone2 sipTrunkGroup SIPREC_TG2 media mediaIpInterfaceGroupName IG2
set addressContext default zone zone2 sipTrunkGroup SIPREC_TG2 ingressIpPrefix <IP address> <prefix>
set addressContext default zone zone2 sipTrunkGroup SIPREC_TG2 state enabled
set addressContext default zone zone2 sipTrunkGroup SIPREC_TG2 mode inService
commit
```

SBC Configurations for SIPRec

We mustmakea separate TG with separate zone and sipSigport and attach that to egress IP interface group. This sip trunk is toward NICE recorder.

1. Create new Zone and Configure Sip Sigport for SIPRec Zone.

```
set addressContext default zone zone4 id 444
set addressContext default zone zone4 sipSigPort 4 ipInterfaceGroupName IG2
set addressContext default zone zone4 sipSigPort 4 ipAddressV4 <IPv4 address>
set addressContext default zone zone4 sipSigPort 4 portNumber <1-65535>
set addressContext default zone zone4 sipSigPort 4 transportProtocolsAllowed sip-udp,sip-tcp,sip-tls-tcp
set addressContext default zone zone4 sipSigPort 4 siprec enabled
set addressContext default zone zone4 sipSigPort 4 mode inService
set addressContext default zone zone4 sipSigPort 4 state enabled
commit
```

2. Configure Trunk group for SIPRec zone.

Please ensure to configure similar transport preferences in CLI and PSX Trunk Group configurations

Also, Transport preference mentioned in SRS Group profile should match transport preferences inTrunk Group towards SIPRec zone.

```
set addressContext default zone zone4 sipTrunkGroup SIPREC_TG4 media mediaIpInterfaceGroupName IG2
set addressContext default zone zone4 sipTrunkGroup SIPREC_TG4 ingressIpPrefix <IP address> <prefix>
set addressContext default zone zone4 sipTrunkGroup SIPREC_TG4 signaling transportPreference preference1 tls-tcp
set addressContext default zone zone4 sipTrunkGroup SIPREC_TG4 state enabled
set addressContext default zone zone4 sipTrunkGroup SIPREC_TG4 mode inService
commit
```

3. The Path Check Profilespecifies the conditions that constitute a connectivity failure, and in the event of such a failure, the conditions that constitute a connectivity recovery.This profile specifies the configuration for OPTIONS PING.

```
set profiles services pathCheckProfile sip_recording1 protocol sipOptions
set profiles services pathCheckProfile sip_recording1 sendInterval 10
set profiles services pathCheckProfile sip_recording1 replyTimeoutCount 3
set profiles services pathCheckProfile sip_recording1 recoveryCount 1
set profiles services pathCheckProfile sip_recording1 failureResponseCodes [ all5xx ]
set profiles services pathCheckProfile sip_recording1 transportPreference preference1 tls-tcp
```

4. Configure the SRS IP as an ipPeer in the SIPREC zone (the zone containing the Trunk Group configured for the SRS) and attach the pathcheck profile to it.

```
set addressContext default zone zone4 ipPeer SIPREC_VRSP1 ipAddress <The IPv4 or IPv6 address of the Peer>
set addressContext default zone zone4 ipPeer SIPREC_VRSP1 ipPort <0-65535>
set addressContext default zone zone4 ipPeer SIPREC_VRSP1 pathCheck profile sip_recording1
set addressContext default zone zone4 ipPeer SIPREC_VRSP1 pathCheck state enabled
set addressContext default zone zone4 ipPeer SIPREC_VRSP2 ipAddress <The IPv4 or IPv6 address of the Peer>
set addressContext default zone zone4 ipPeer SIPREC_VRSP2 ipPort <0-65535>
set addressContext default zone zone4 ipPeer SIPREC_VRSP2 pathCheck profile sip_recording1
set addressContext default zone zone4 ipPeer SIPREC_VRSP2 pathCheck state enabled
commit
```

5. NICE does not support SIP INFO method towards SIPRec . So, disable SIP INFO method towards SIPRec Trunk Group.

```
set addressContext default zone zone4 sipTrunkGroup SIPREC_TG4 signaling methods info reject
commit
```

5.CreatesipRecMetadataProfile with version 1 as per RFC 7865 and associate the profile to SIPRec Trunk Group.

> When sipRecMetadataProfile is not configured,by defaultSBC supports backward compatibility and pre-defined metadata for passing proprietary call specific information from the SRC to the SRS.

Refer toMetadataSupportfor additional NICE configurations.

```
set profiles services sipRecMetadataProfile t1 state enabled
set profiles services sipRecMetadataProfile t1 version 1
comm
set addressContext default zone zone4 sipTrunkGroup SIPREC_TG4 services sipRecMetadataProfile t1
comm
```

TLS Certificates

The Public Key Infrastructure (PKI) provides a common set of infrastructure features supporting public key and certificate-based authentication based on the RSA public/private key pairs and X.509 digital certificates.

Import all the required certificated to SBC under /opt/sonus/external and execute the following commands.

```
####   SRS1 Application Server Certificate Import  ####
set system security pki certificate NICE_REMOTE1 state enabled
set system security pki certificate NICE_REMOTE1 fileName <SRS1 Certficate filename imported in SBC>
set system security pki certificate NICE_REMOTE1 type remote
comm

####   SRS2 Interaction Server Certificate Import  ####
set system security pki certificate NICE_REMOTE2 state enabled
set system security pki certificate NICE_REMOTE2 fileName <SRS2 Certficate filename imported in SBC>
set system security pki certificate NICE_REMOTE2 type remote
comm

#### SBC Certificate Import  ####
set system security pki certificate SBC_LOCAL state enabled
set system security pki certificate SBC_LOCAL fileName <SBC local Certficate filename imported in SBC>
set system security pki certificate SBC_LOCAL passPhrase xxxx
set system security pki certificate SBC_LOCAL type local
comm
```

TLS Profile

This object creates and configures a profile for implementing the Transport Layer Security (TLS) protocol to use with SIP over TLS. TLS is an IETF protocol for securing communications across an untrusted network. Normally, SIP packets travel in plain text over TCP or UDP connections. Secure SIP is a security measure that uses TLS, the successor to the Secure Sockets Layer (SSL) protocol.

To add a TLS protection-level policy, create a TLS PROFILE and configure each of the parameters.

```
####        TLS Profile for SIP Endpoint        ####
set profiles security tlsProfile TLS_SIPREC1 appAuthTimer 5
set profiles security tlsProfile TLS_SIPREC1 handshakeTimer 5
set profiles security tlsProfile TLS_SIPREC1 sessionResumpTimer 3600
set profiles security tlsProfile TLS_SIPREC1 cipherSuite1 rsa-with-aes-128-cbc-sha
set profiles security tlsProfile TLS_SIPREC1 cipherSuite2 rsa-with-aes-256-cbc-sha
set profiles security tlsProfile TLS_SIPREC1 cipherSuite3 tls_rsa_with_aes_256_gcm_sha384
set profiles security tlsProfile TLS_SIPREC1 allowedRoles clientandserver
set profiles security tlsProfile TLS_SIPREC1 authClient true
set profiles security tlsProfile TLS_SIPREC1 clientCertName SBC_LOCAL
set profiles security tlsProfile TLS_SIPREC1 serverCertName SBC_LOCAL
set profiles security tlsProfile TLS_SIPREC1 acceptableCertValidationErrors none
set profiles security tlsProfile TLS_SIPREC1 v1_0 enabled
set profiles security tlsProfile TLS_SIPREC1 v1_1 enabled
set profiles security tlsProfile TLS_SIPREC1 v1_2 enabled
set profiles security tlsProfile TLS_SIPREC1 suppressEmptyFragments disabled
set profiles security tlsProfile TLS_SIPREC1 peerNameVerify disabled
commit


####        TLS Profile for NICE SIP Recording Trunk        ####
set profiles security tlsProfile testsiprectlsroot appAuthTimer 5
set profiles security tlsProfile testsiprectlsroot handshakeTimer 5
set profiles security tlsProfile testsiprectlsroot sessionResumpTimer 3600
set profiles security tlsProfile testsiprectlsroot cipherSuite1 rsa-with-aes-128-cbc-sha
set profiles security tlsProfile testsiprectlsroot cipherSuite2 rsa-with-aes-256-cbc-sha
set profiles security tlsProfile testsiprectlsroot cipherSuite3 tls_rsa_with_aes_256_gcm_sha384
set profiles security tlsProfile testsiprectlsroot allowedRoles clientandserver
set profiles security tlsProfile testsiprectlsroot authClient true
set profiles security tlsProfile testsiprectlsroot clientCertName SBC_LOCAL
set profiles security tlsProfile testsiprectlsroot serverCertName SBC_LOCAL
set profiles security tlsProfile testsiprectlsroot acceptableCertValidationErrors none
set profiles security tlsProfile testsiprectlsroot v1_0 enabled
set profiles security tlsProfile testsiprectlsroot v1_1 enabled
set profiles security tlsProfile testsiprectlsroot v1_2 enabled
set profiles security tlsProfile testsiprectlsroot suppressEmptyFragments disabled
set profiles security tlsProfile testsiprectlsroot peerNameVerify disabled
commit
```

The TLS profile is specified on the SIP Signaling Port and controls behavior of all TLS connections established on that signaling port.

```
###### Attach TLS profile to SIPrec zone ######
set addressContext default zone zone4 sipSigPort 4 tlsProfileName testsiprectlsroot
comm

###### Attach TLS profile to SIPrec zone (If TLS transport is enabled)######
set addressContext default zone zone1 sipSigPort 1 tlsProfileName TLS_SIPREC1
set addressContext default zone zone2 sipSigPort 2 tlsProfileName TLS_SIPREC1
comm
```

SBC Configuration to enable PSX

We need to disable local PolicyServer and configure remote PSX details in SBC SWe Core.

```
set system policyServer localServer PSX_LOCAL_SERVER state disabled
set system policyServer localServer PSX_LOCAL_SERVER mode outOfService
set system policyServer remoteServer IOTPSX ipAddress 172.16.100.216
set system policyServer remoteServer IOTPSX state enabled
set system policyServer remoteServer IOTPSX mode active
set system policyServer remoteServer IOTPSX action force
commit
```

PSX Configurations forRibbonSBC SWe Core

Configuring Class of Service

Please note that we have used default Class Of Service 'DEFAULT_IP' for our testing.

Figure 5:

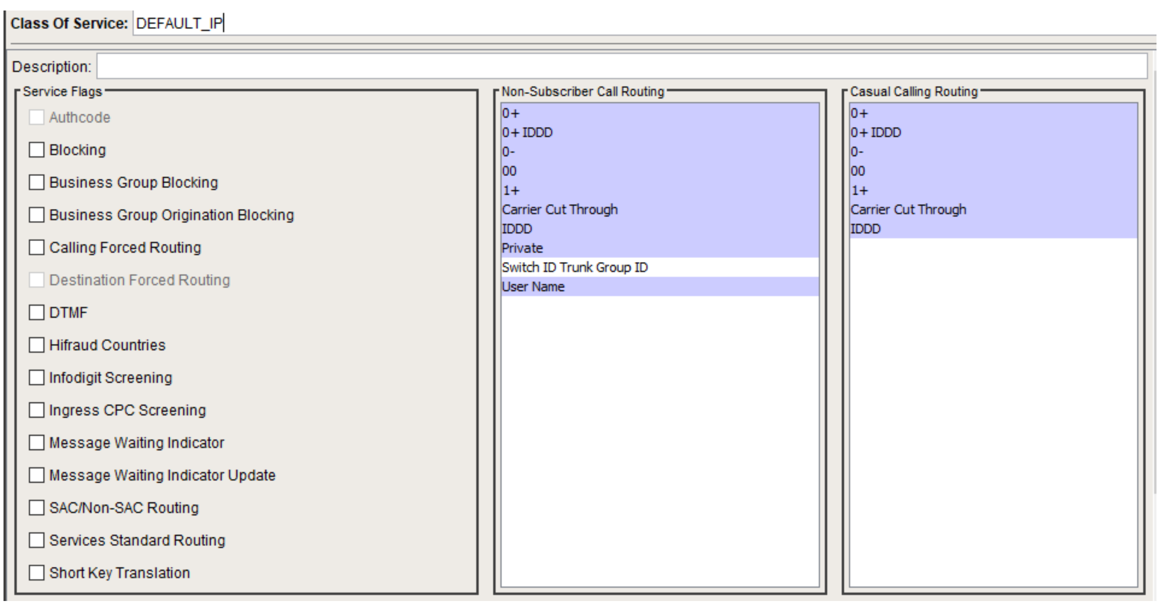
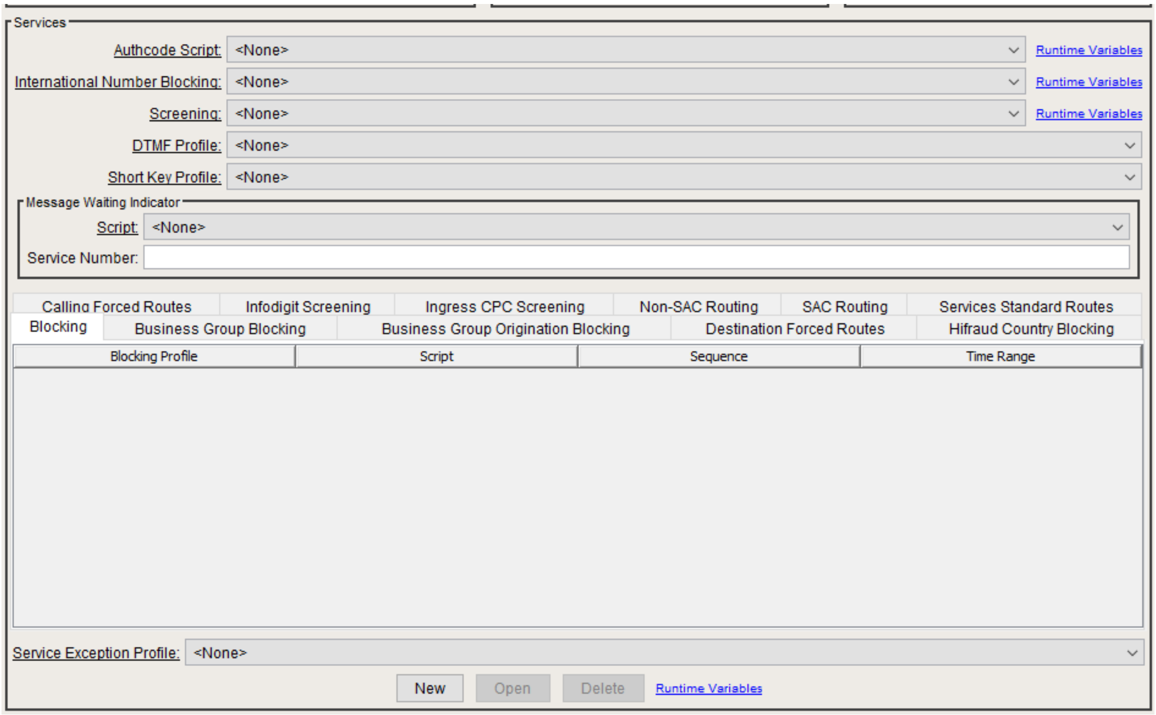


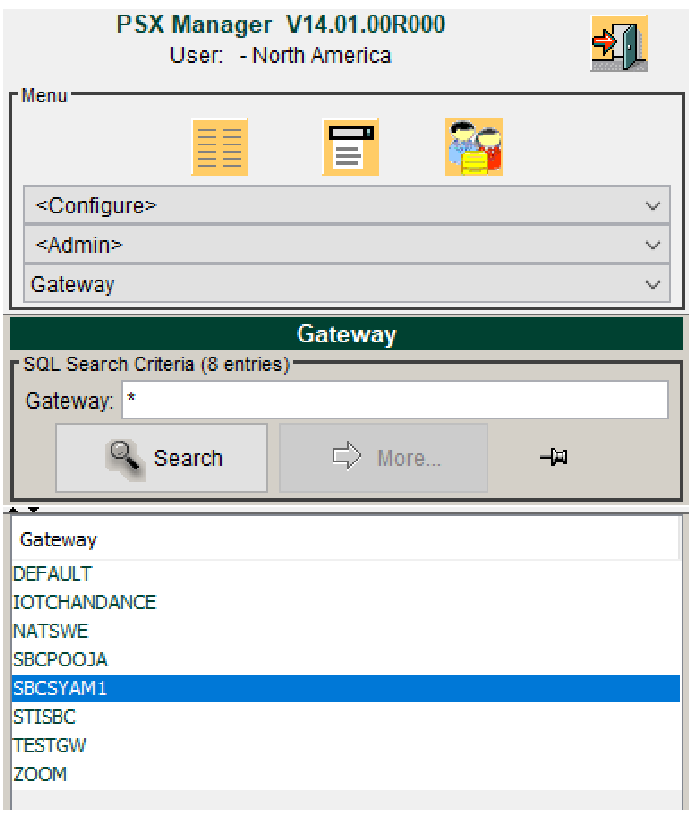
Figure6:



Configuring Gateway

1.Configure a gateway with the SBC name and the management IP address.

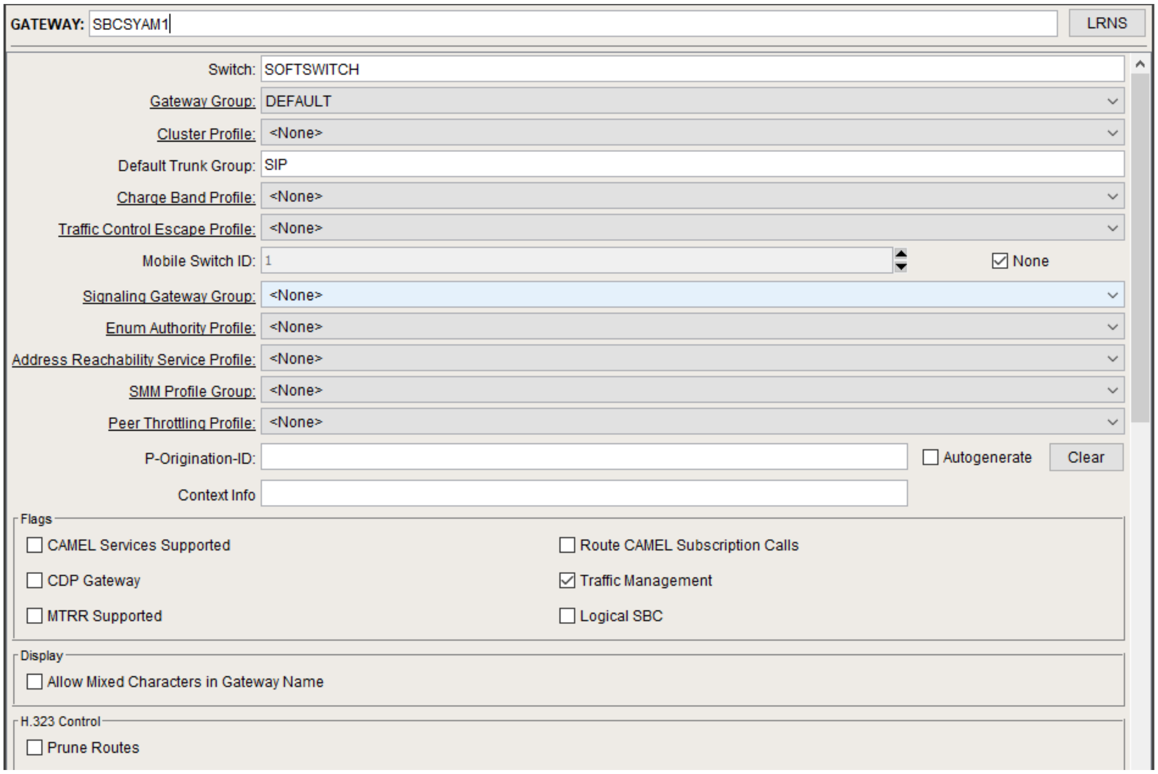
Figure 7:



2. From the Gateway configuration UI, enter the name of gateway that is configured in the SBC.

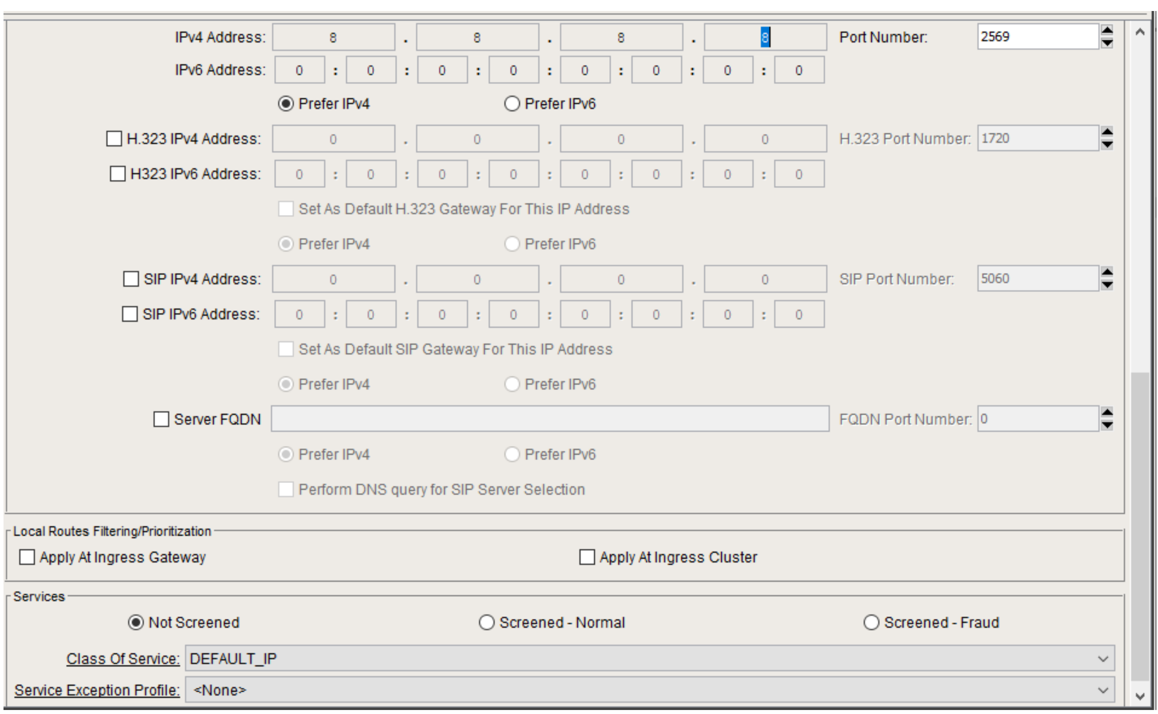
> Gateway name should be same as systemname in SBC conf file and should be capitalized.

Figure8:



Configure SBC management IP in IPv4 Address and default port number 2569.

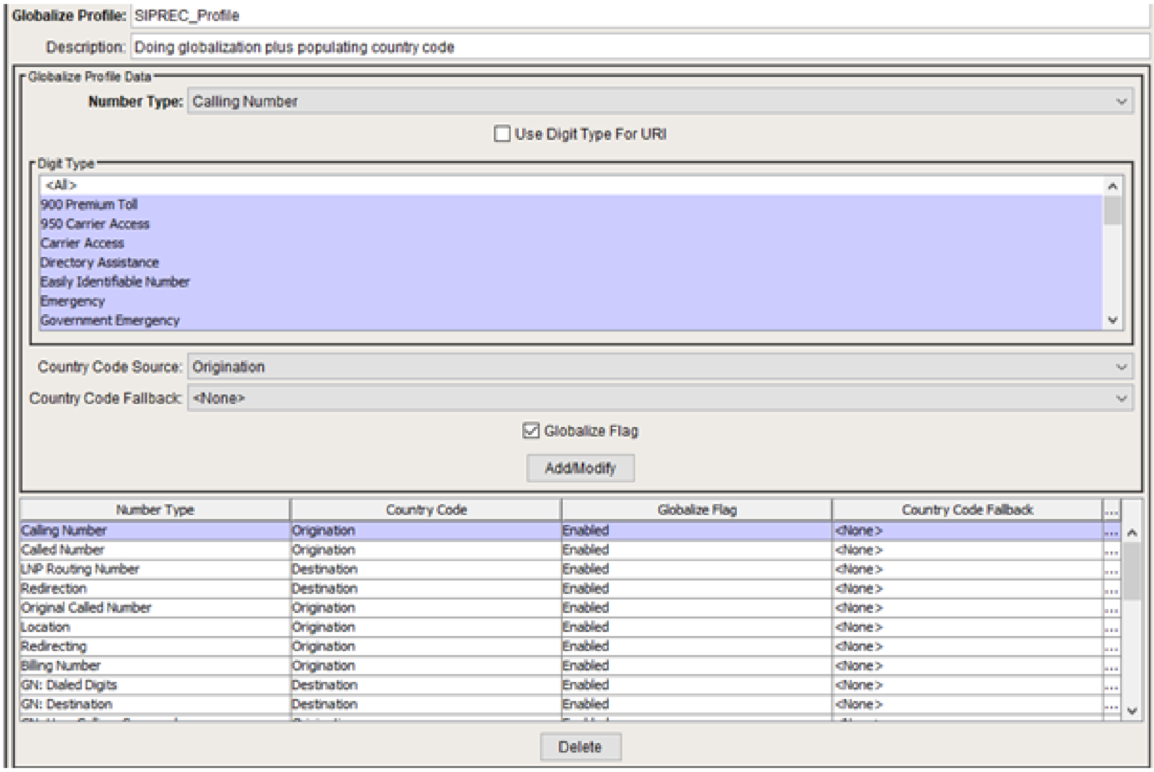
Figure 9:



Configuring Globalization Profile

This object is used to define numbers that are to be globalized for egress SIP message headers. Specify a profile entry for each number type that needs to be globalized. The profile includes a digit type, a source for the country code, and a flag to enable the globalization. Assign Globalize Profiles to egress trunk groups by selecting them on the IP Signaling Profile for each trunk group.

Figure10:


Globalize Profile: SIPREC_Profile

Description: Doing globalization plus populating country code

Globalize Profile Data

Number Type: Calling Number

Use Digit Type For URI

Digit Type

<All>
900 Premium Toll
950 Carrier Access
Carrier Access
Directory Assistance
Easily Identifiable Number
Emergency
Government Emergency

Country Code Source: Origination

Country Code Fallback: <None>

Globalize Flag

Add/Modify

Number Type	Country Code	Globalize Flag	Country Code Fallback
Calling Number	Origination	Enabled	<None>
Called Number	Origination	Enabled	<None>
LNP Routing Number	Destination	Enabled	<None>
Redirection	Destination	Enabled	<None>
Original Called Number	Origination	Enabled	<None>
Location	Origination	Enabled	<None>
Redirecting	Origination	Enabled	<None>
Billing Number	Origination	Enabled	<None>
GN: Dialed Digits	Destination	Enabled	<None>
GN: Destination	Destination	Enabled	<None>

Delete

Figure11:

Number Type	Country Code	Globalize Flag	Country Code Fallback
GN: User Calling, Not Screened	Origination	Enabled	<None>
GN: Redirecting Terminating	Destination	Enabled	<None>
GN: Ported Dialed	Destination	Enabled	<None>
GN: Called CES Id	Destination	Enabled	<None>
GN: Additional Called	Destination	Enabled	<None>
GN: Additional Connected	Destination	Enabled	<None>
GN: Additional Calling	Origination	Enabled	<None>
GN: Additional Original Called	Destination	Enabled	<None>
GN: Additional Redirecting	Origination	Enabled	<None>
GN: Additional Redirection	Destination	Enabled	<None>
Contractor Number	Origination	Enabled	<None>

Figure12:

Number Type	Country Code	Globalize Flag	Country Code Fallback
GN: Additional Calling	Origination	Enabled	<None>
GN: Additional Original Called	Destination	Enabled	<None>
GN: Additional Redirecting	Origination	Enabled	<None>
GN: Additional Redirection	Destination	Enabled	<None>
Contractor Number	Origination	Enabled	<None>
GN: Network Provided Number	Destination	Enabled	<None>
Dialed Number	Origination	Enabled	<None>
GN: Third Party Number	Destination	Enabled	<None>
GN: Collect Call Number	Destination	Enabled	<None>
GN: Local ANI	Origination	Enabled	<None>
To URI User	Origination	Enabled	<None>

ConfiguringIP SignalingProfile

This object specifies parameters associated with H.323, SIP, SIP-I communication that are sent as part of the outgoing signaling message after standard protocol rules have been applied.

You can associate IP signaling profiles with IP trunk groups and virtual trunk groups.

Figure13:

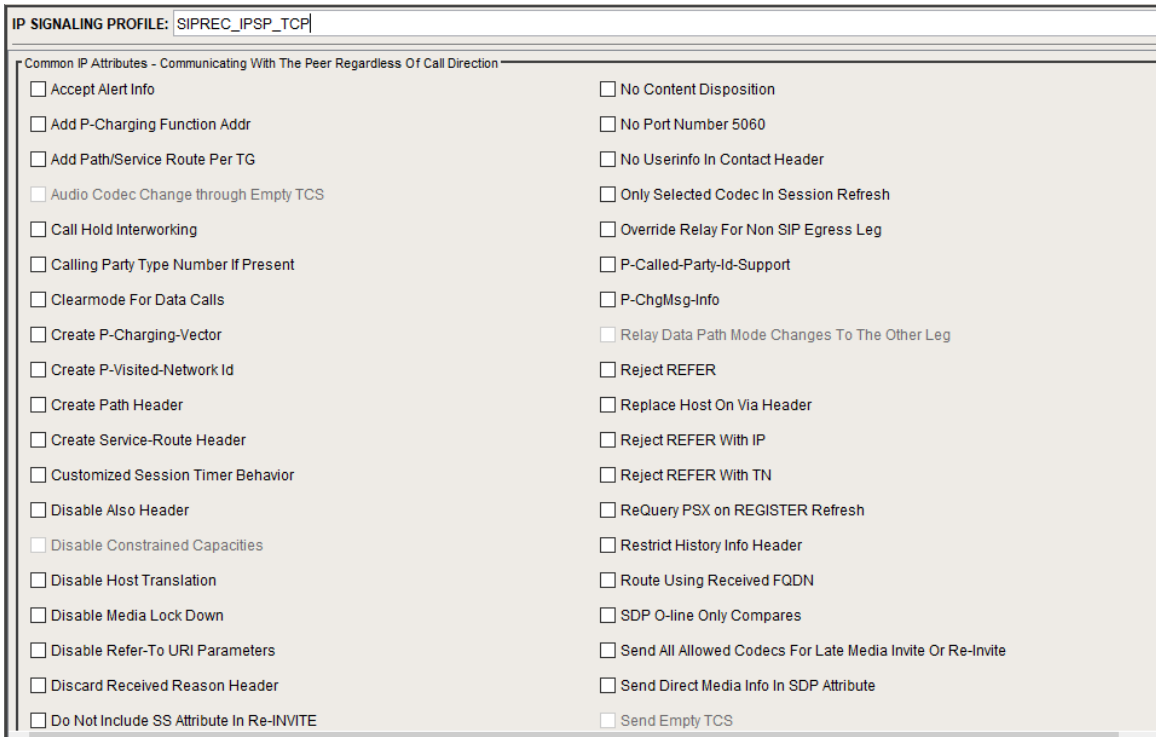


Figure14:

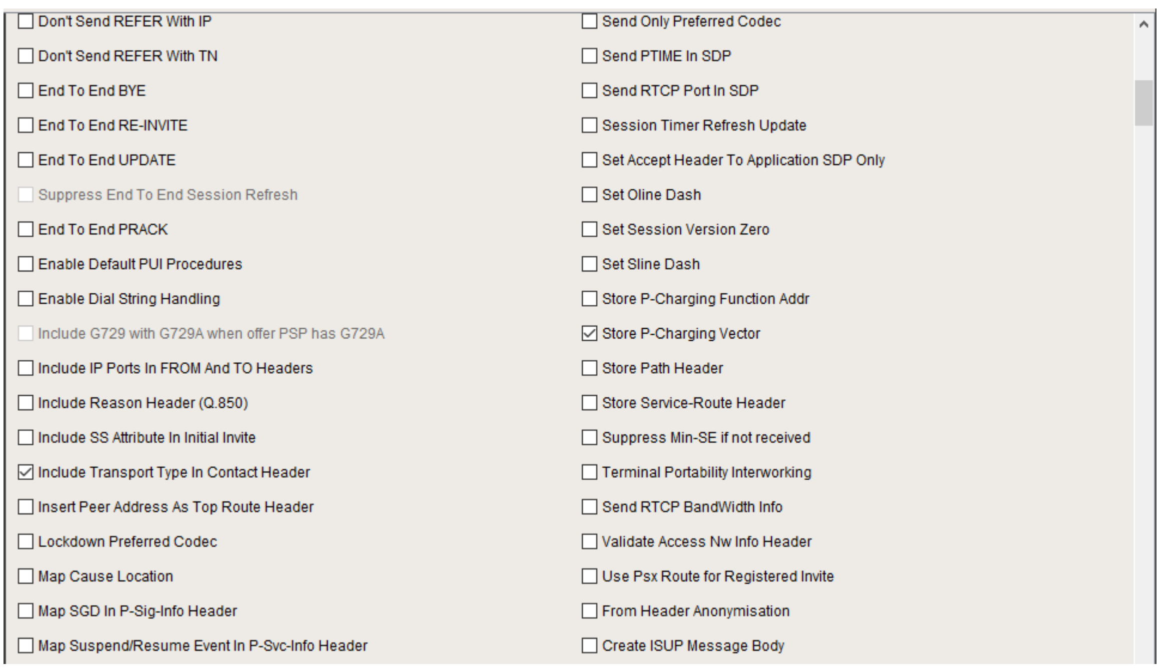


Figure15:

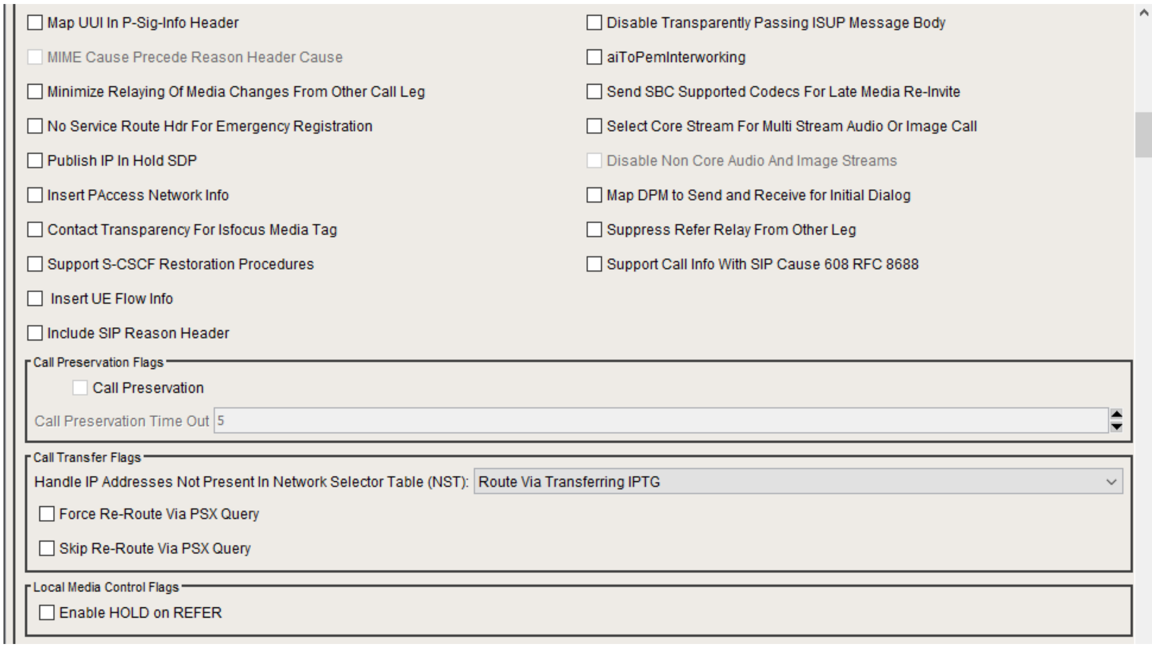


Figure16:

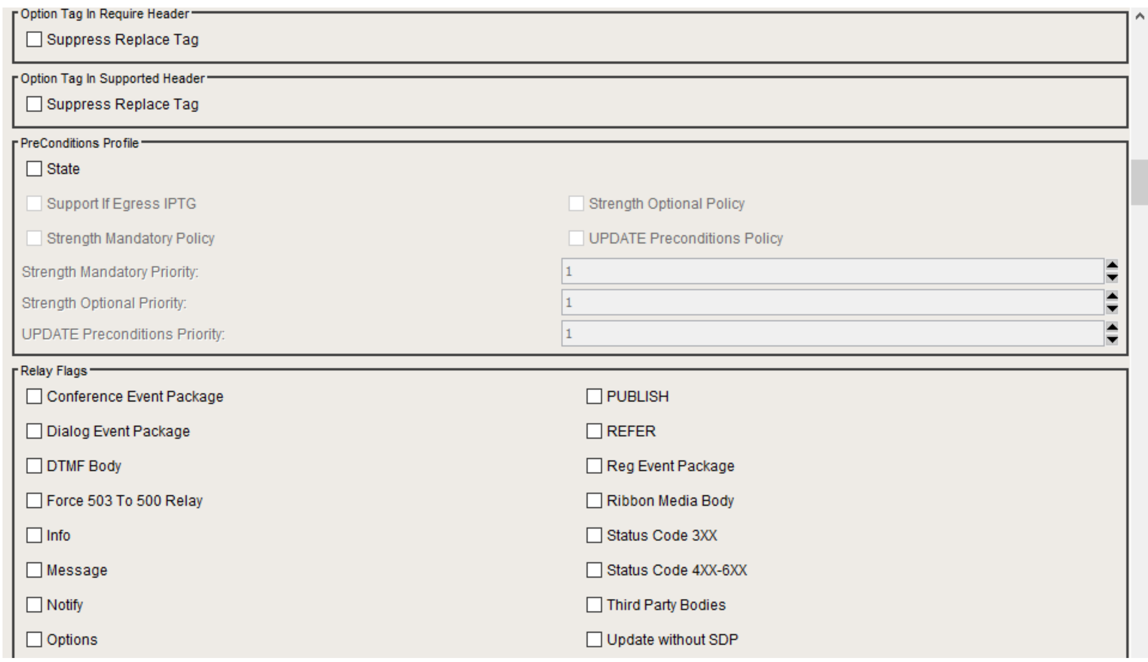


Figure17:

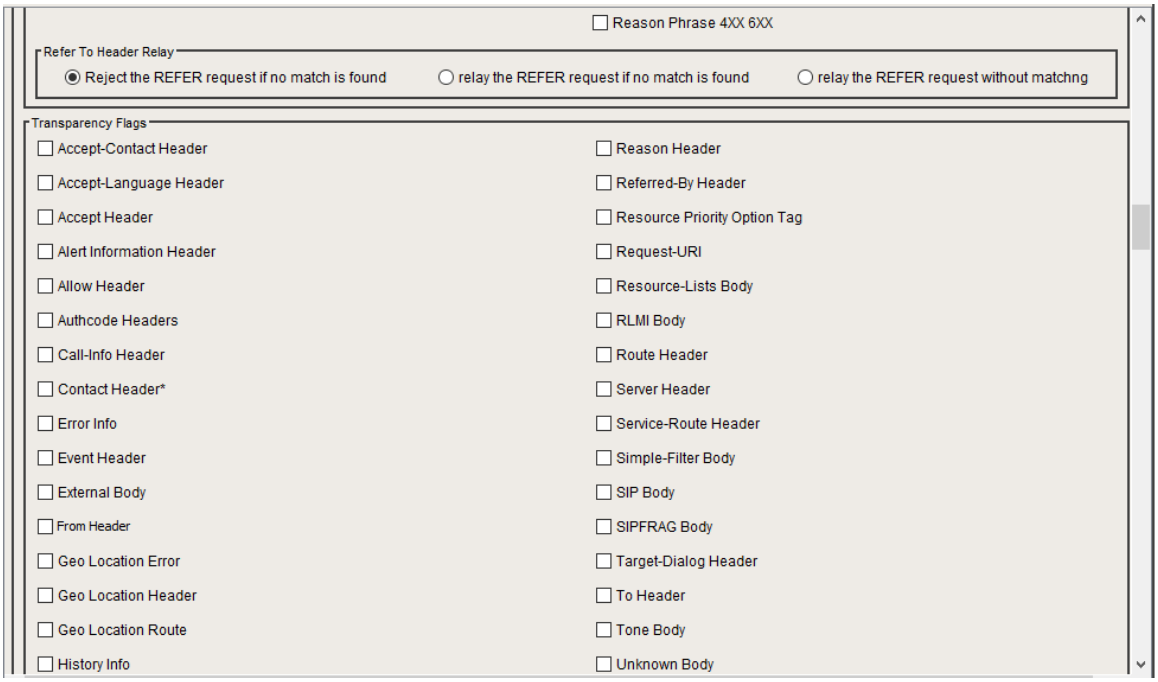
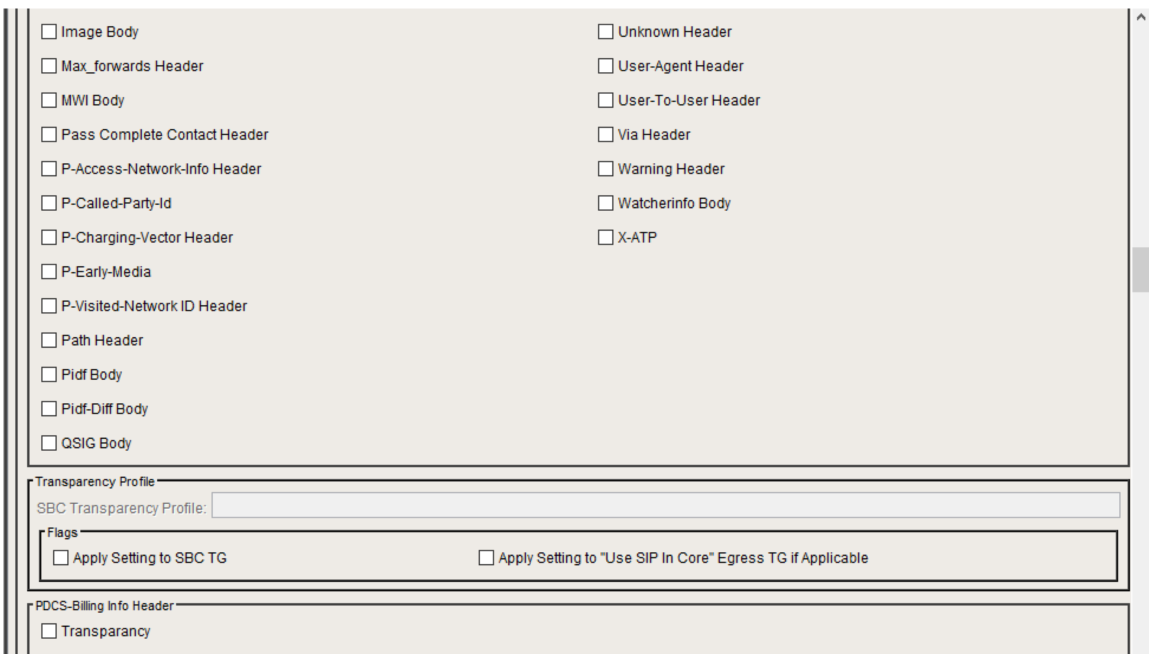


Figure18:



From the drop down, select Globalization Profile created above.

Figure19:

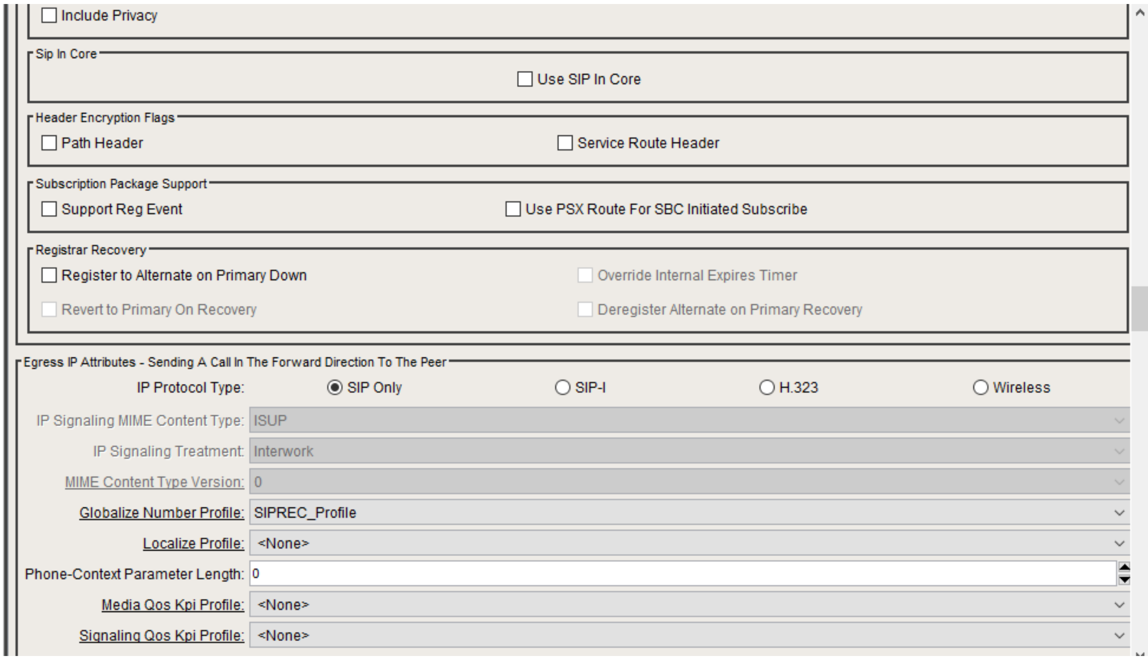


Figure20:

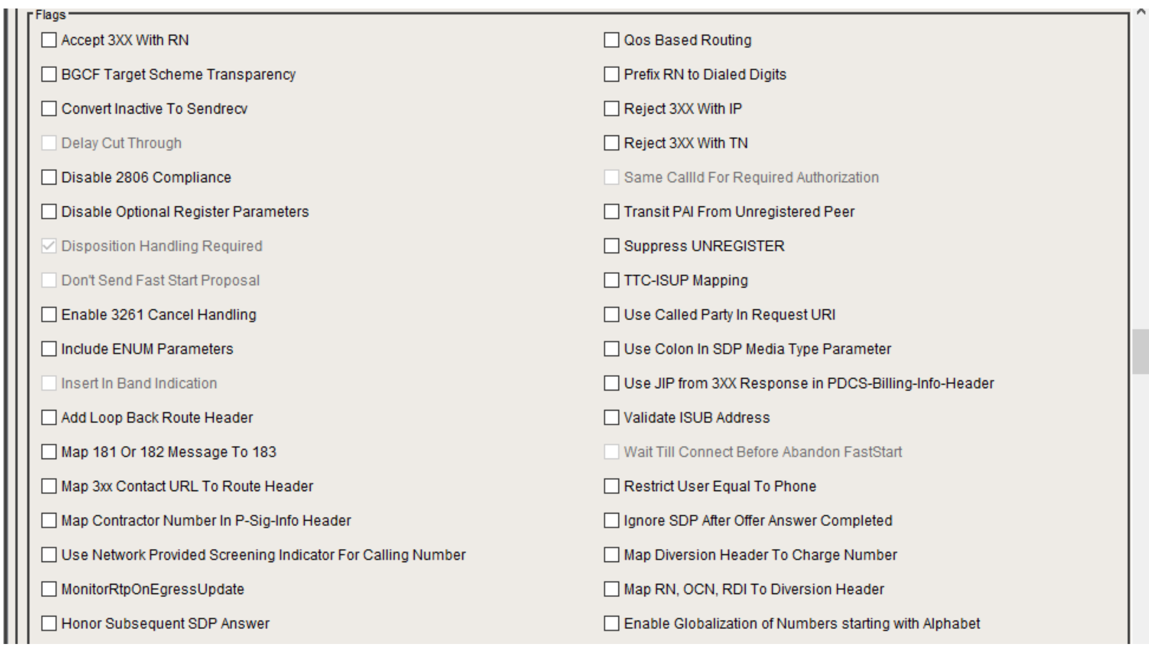


Figure21:

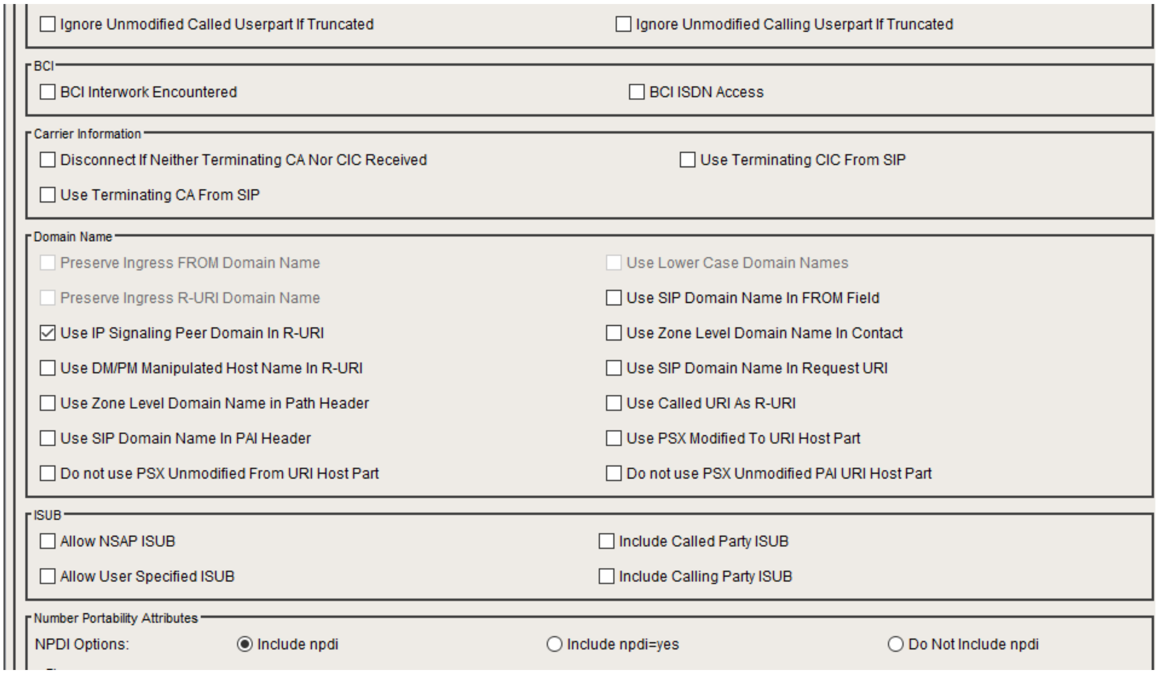


Figure22:

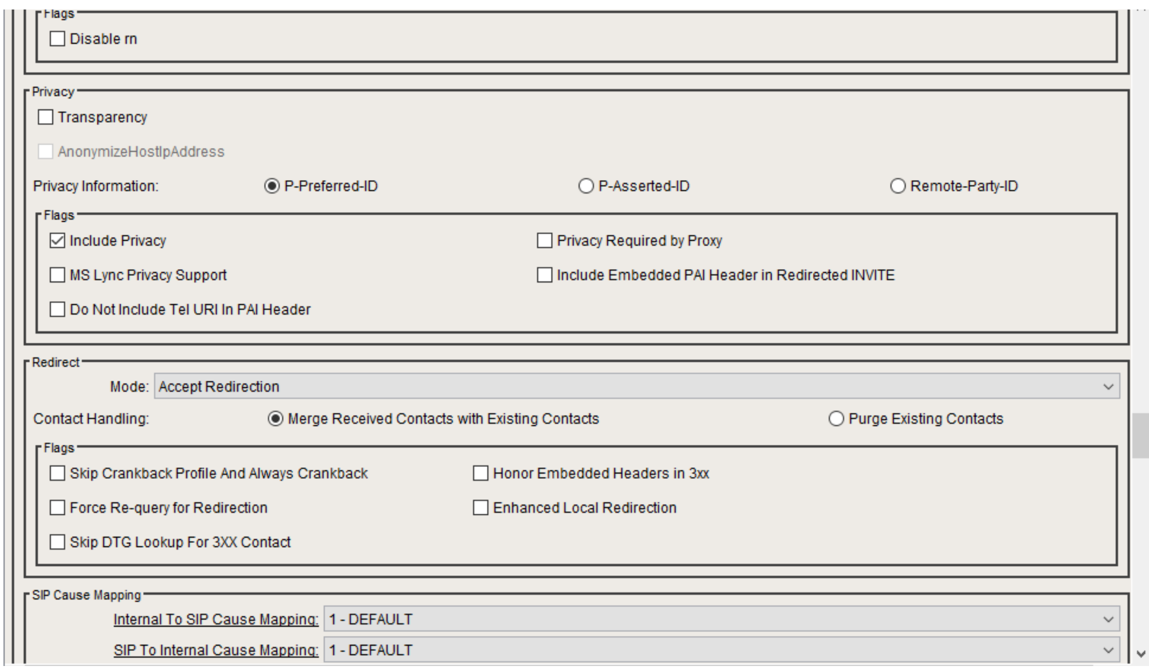


Figure23:

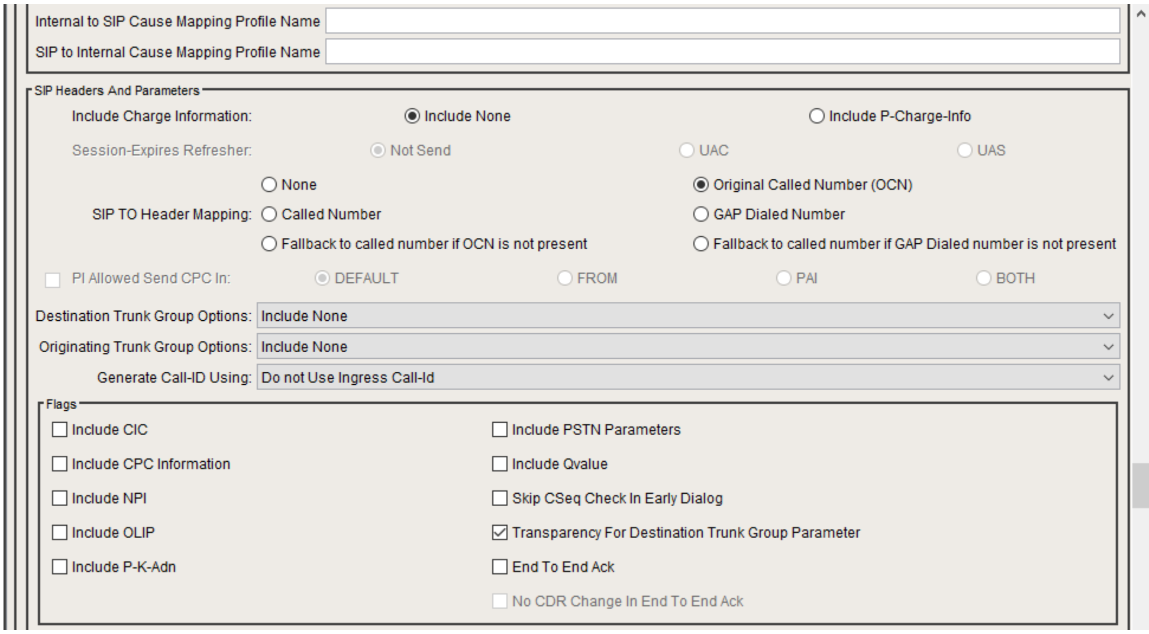
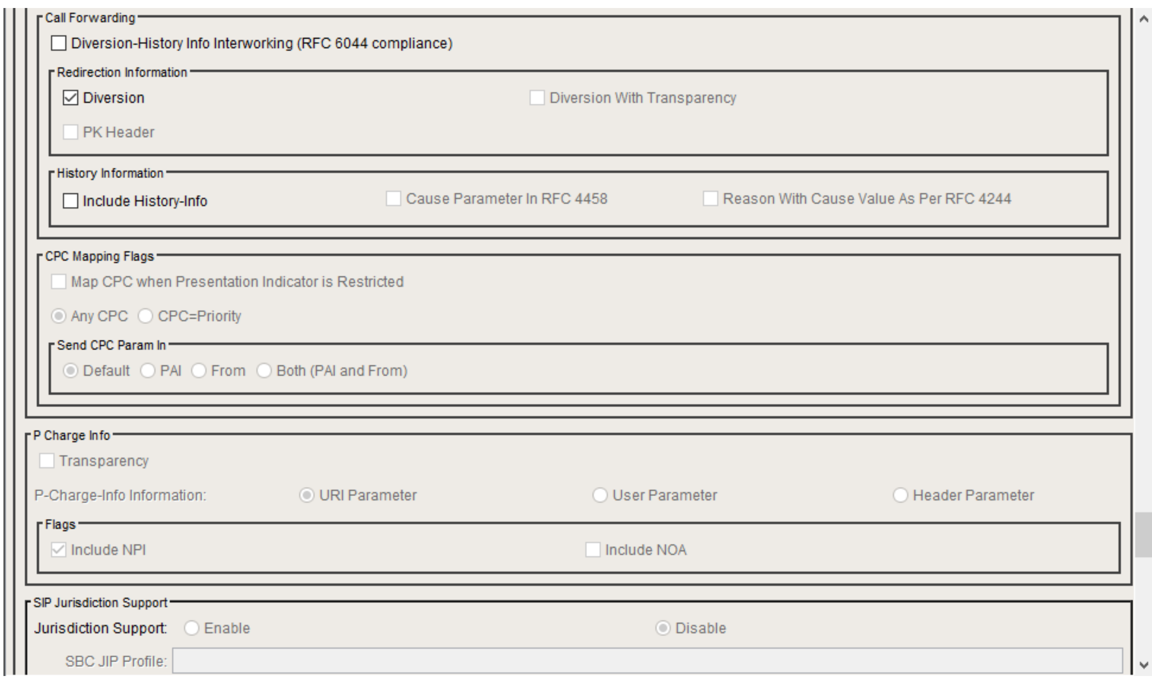


Figure24:



Use Transport Type object to configure the preferred transport.

Figure 25:

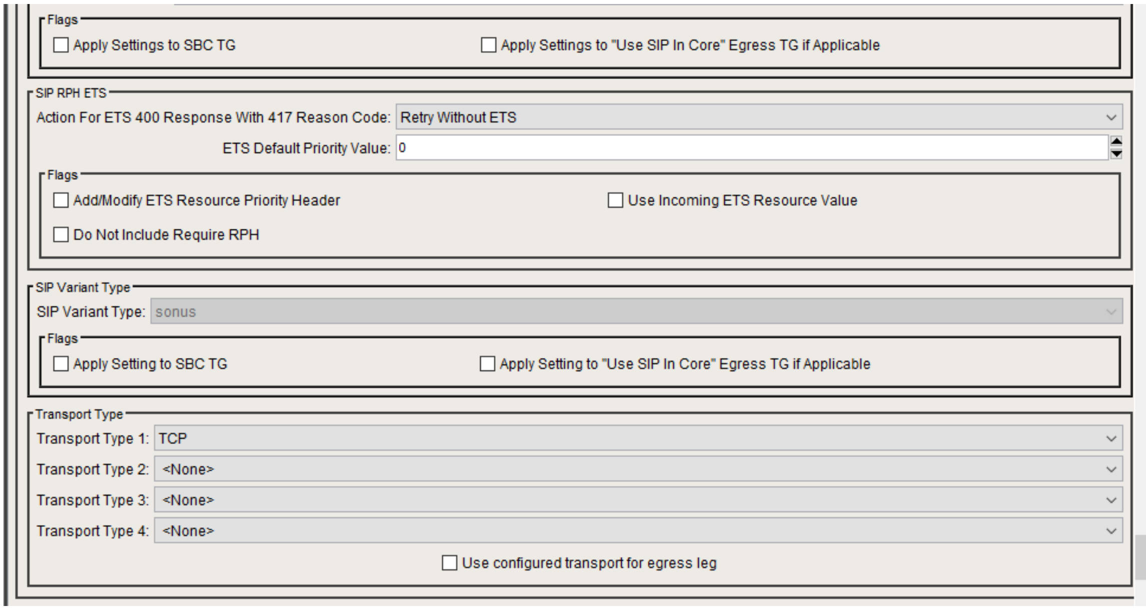


Figure26:

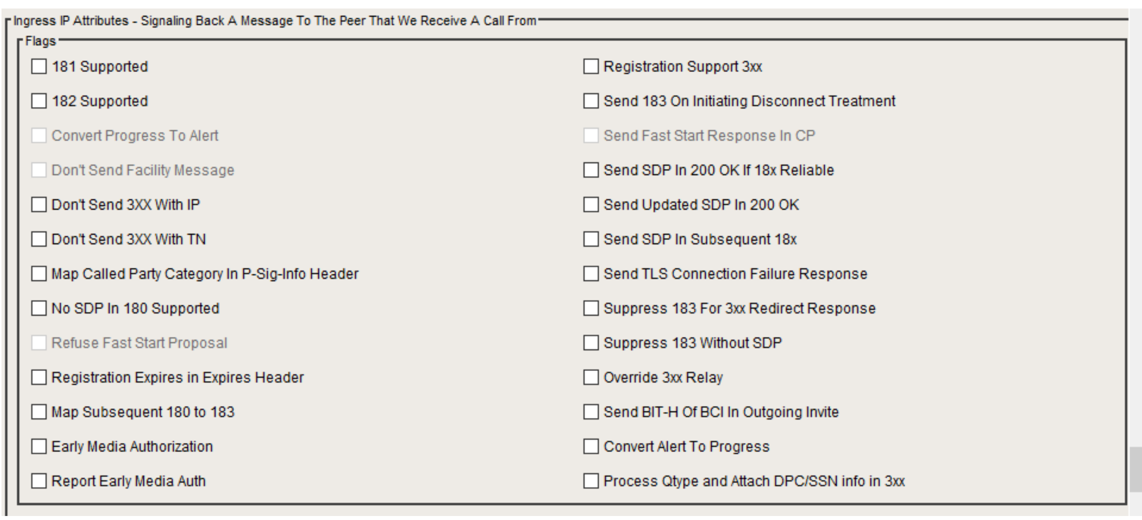
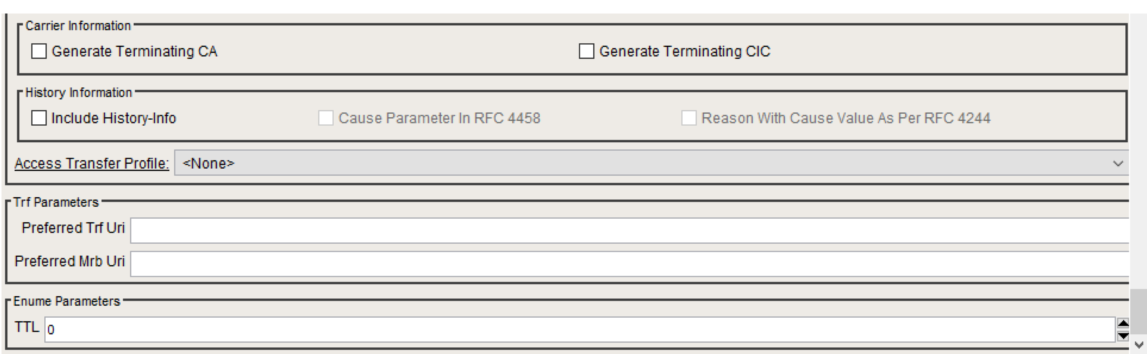


Figure27:



Configuring Codec Entry Profile

Codecs define the audio encoding methods and their associated attributes. You can add custom codec entries which are then available to include when configuring codecs in a Packet Service Profile. When you add a codec entry, the parameters available change, depending on the base codec you select. You can also configure options for a selected Codec Entry that specify how to handle DTMF digits in the media stream.

Define the codec entry priorities and codec names.

DTMF Types Configuration

Use the DTMF relay window under Codec Entry configured in Packet Service Profiles to specifyhow to handle DTMF digits in the media stream.

Figure28:

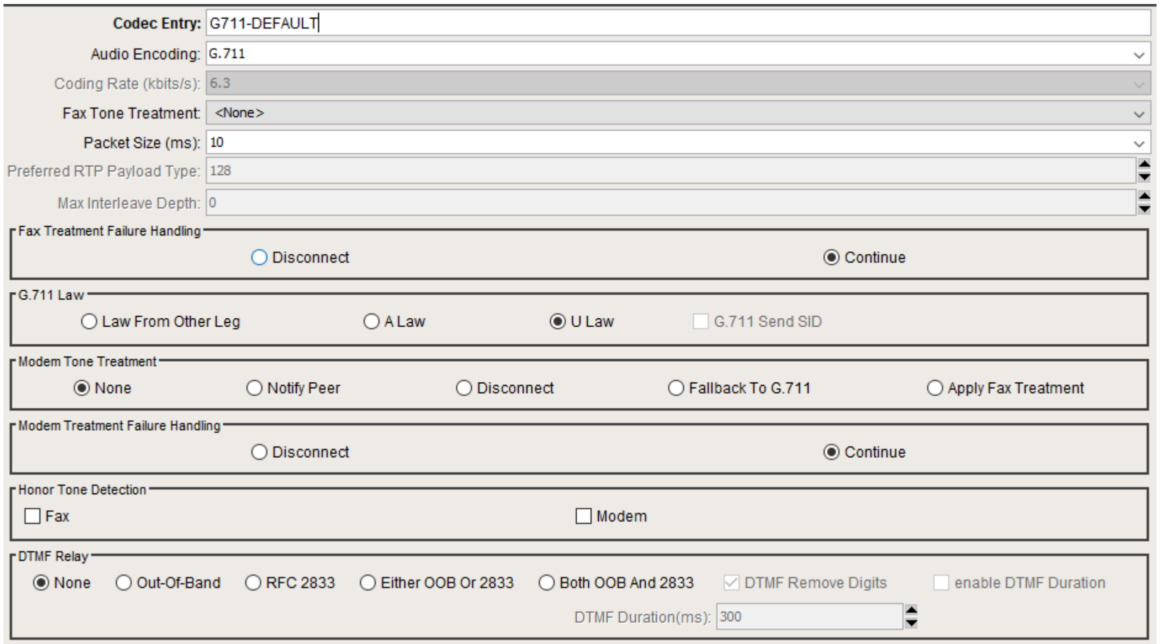


Figure29:

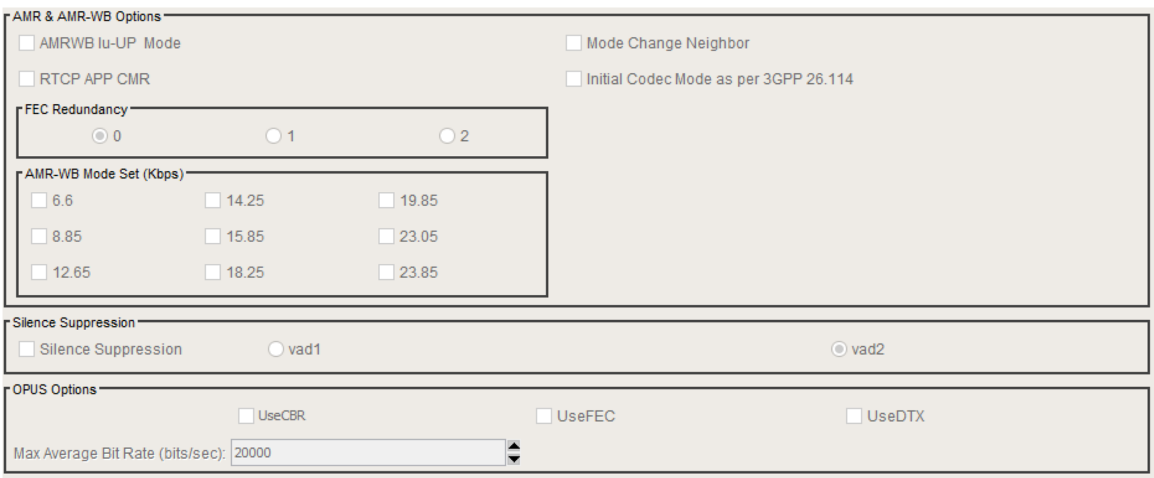
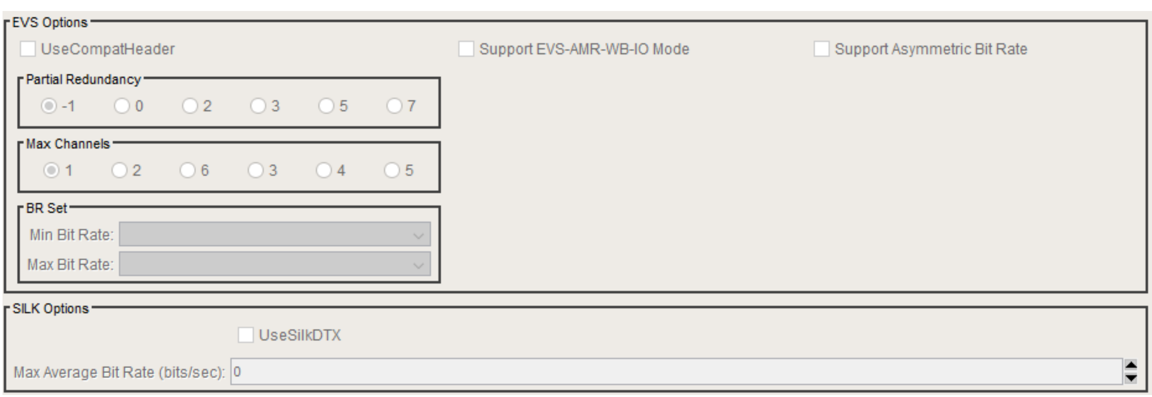


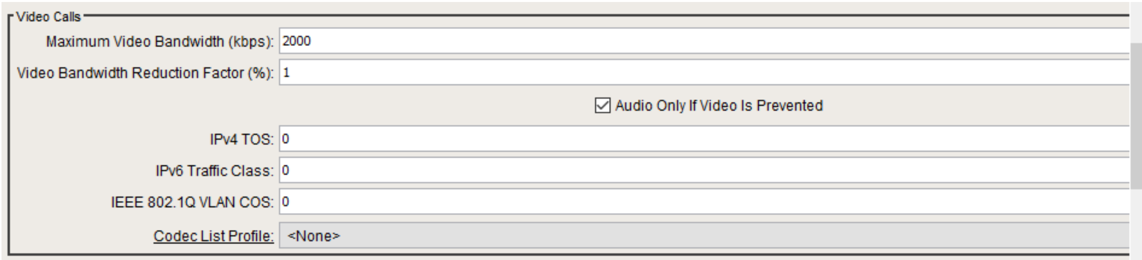
Figure30:



Video Call Configuration

Configure Maximum Video Bandwidth and Video Bandwidth Reduction Factor in packet service profile to enable video calls.

Figure 44:



Configuring Packet Service Profile

Each Packet Service Profile is configured for a pair of gateways, and includes entries for up to four audio/video encoding methods. The pair of gateways can be originating for destination gateways in the same gateway group, or can be originating for destination gateways in an inter-gateway group.

Packet Service Profile IN

From the Drop Down, select the codec Entry profiles created during initial steps,

Figure 31:

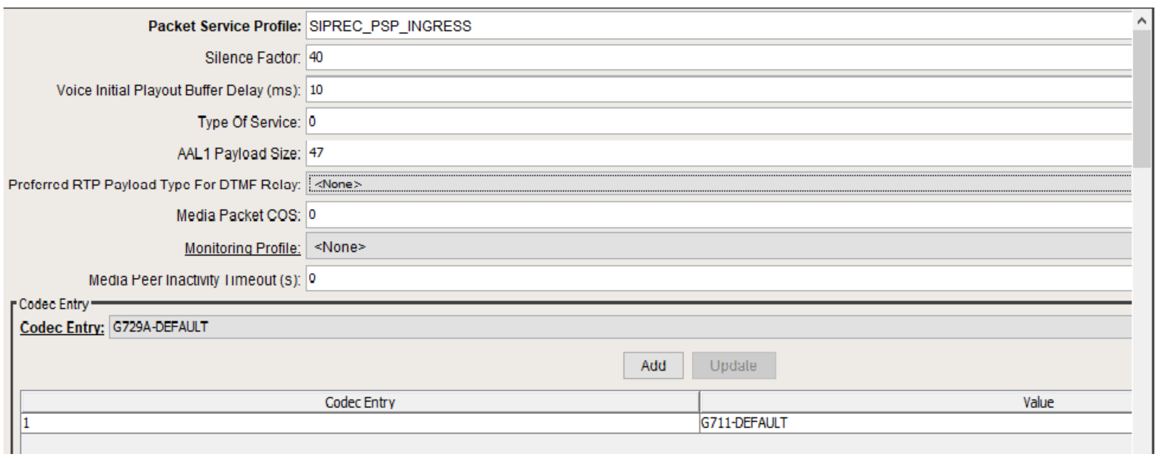
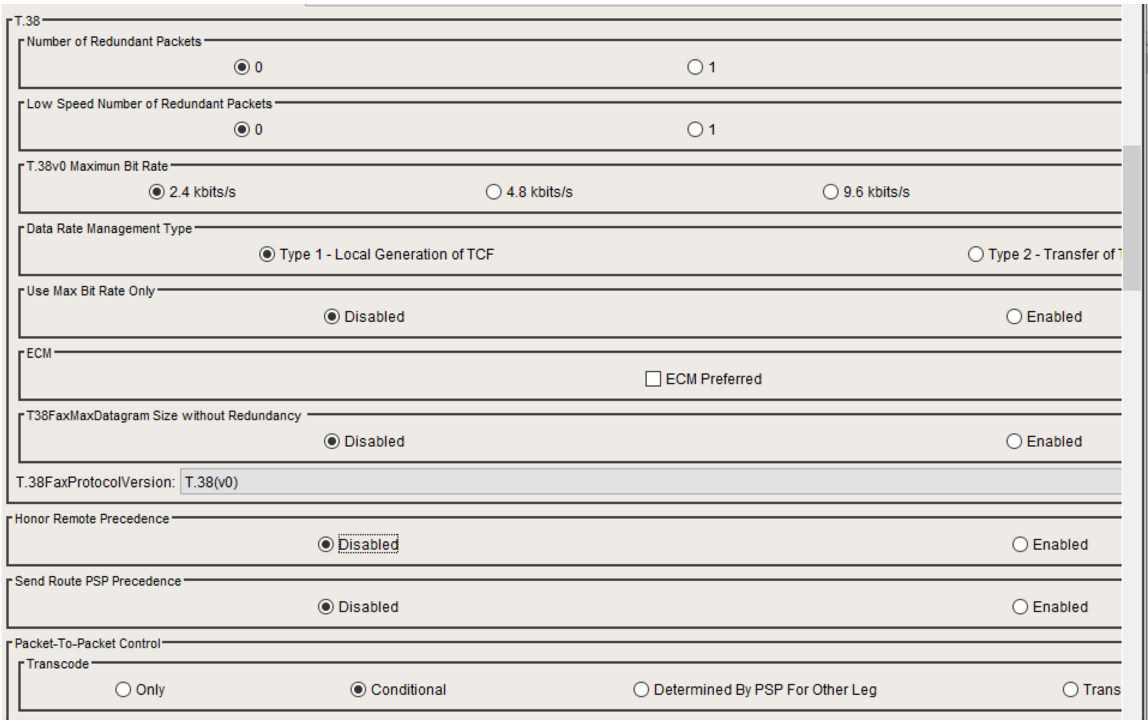


Figure32:



Transcoding:

Use theCodecs Allowed For Transcoding window to specify, for a Packet Service profile (PSP), between which codecs you want the SBC to allow transcoding. Checking options on this window specifies that the codecs selected in the "This Leg" row can be transcoded to those selected in the "Other Leg" row, and vice versa.

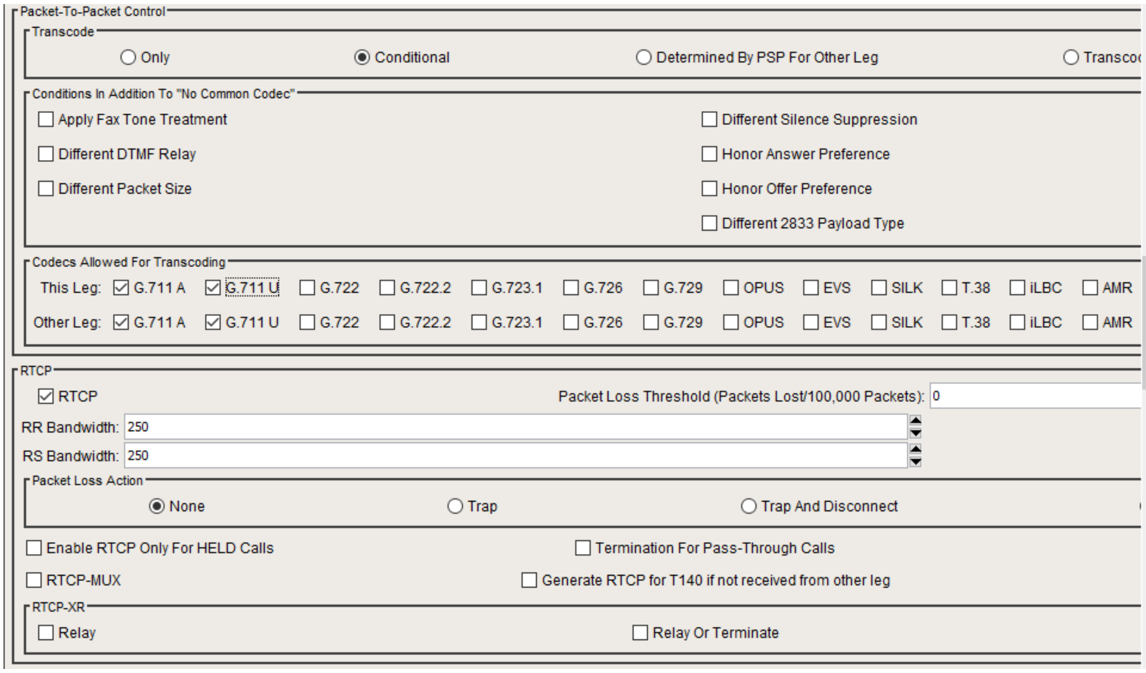
PSPs are assigned to both legs of a call. Therefore the Codecs Allowed For Transcoding values applied to a particular call reflect the contributions of both profiles, with the ingress and egress call legs being viewed as "This Leg" by one profile and as the "Other Leg" by the other profile.

This control specifies the transcoding method used for the associated packet flow.

TheSBC performs transcoding for media streams carried between two IP devices by translating the streams from the ingress audio encoding format to the egress audio encoding format when the devices do not share a common codec. In some environments, transcoding may be preferred over negotiating the attributes of a common codec.

- `Conditional`The SBCperforms transcoding when any of the conditions specified in the Conditions In Addition To "No Common Codec" section are met.
- `Determined By PSP For Other Leg`The SBC performs transcoding based on the transcoding options specified in the packet service profile assigned to the other leg of the call. When selected, PSX Manager disables the check boxes in the Codecs Allowed for Transcoding section.
- `Only`The SBC performs transcoding for the codecs selected in the Codec Allowed For Transcoding section (see definition below). None of the conditions specified in the Conditions In Addition To "No Common Codec" section are used in determining when to perform transcoding.
- `Transcoder Free Transparency`When selected, the SBC transparently passes the PSP from the ingress call-leg to the egress call-leg, bypassing transcoding.

Figure33:



RTCP configuration:

Use this object to specify Real Time Control Protocol (RTCP) options for the call. RTCP is used to report network traffic congestion data.

When set to**Enable**, Use RTCP for the call leg.

Figure34:

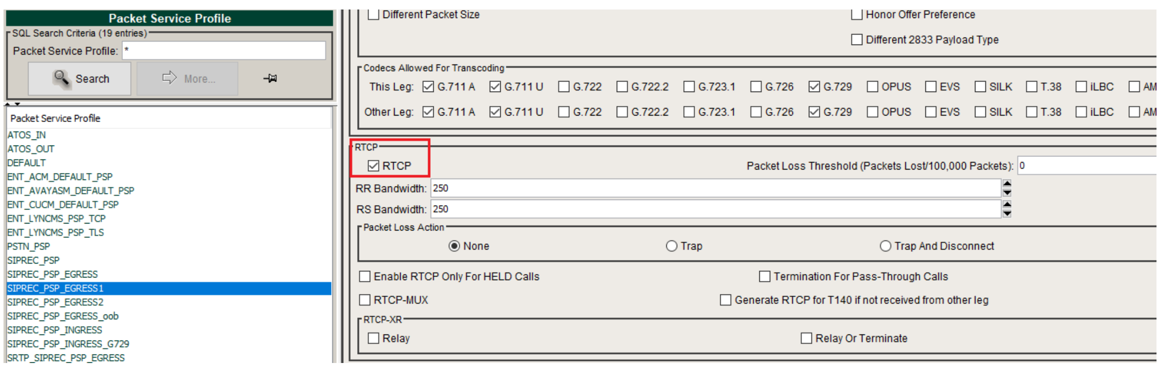
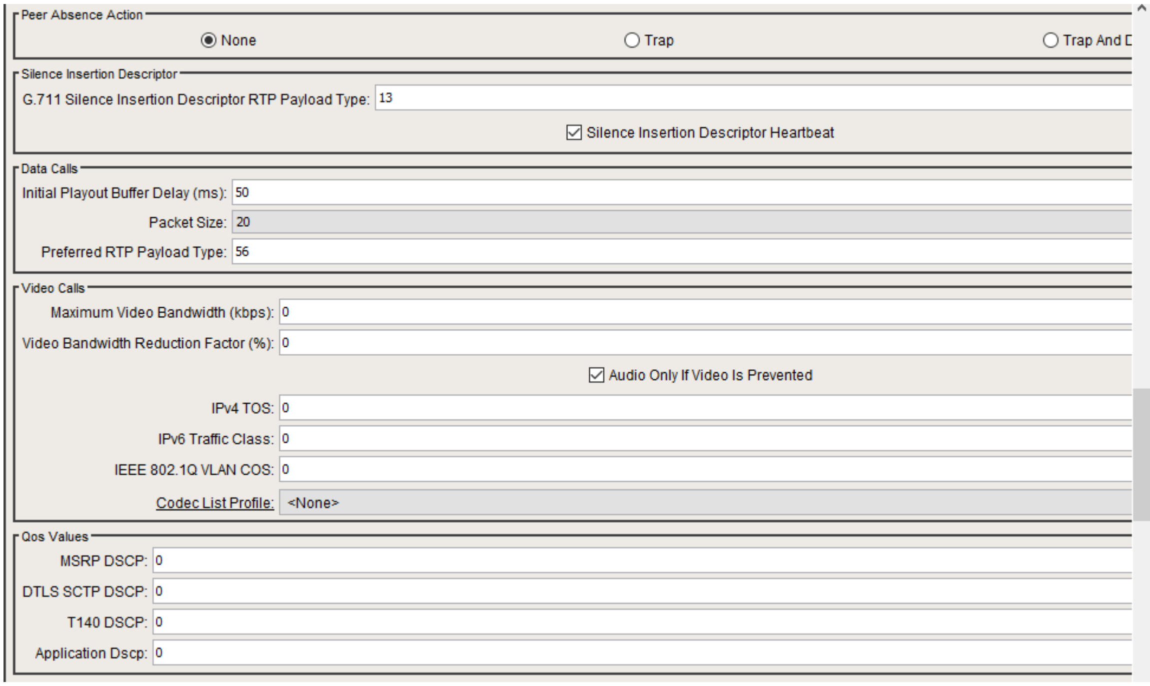


Figure35:



Secure RTP/RTCP > Crypto Suite Profile is used for srtp configurations. Please refer Media Encryption for more details

Figure36:

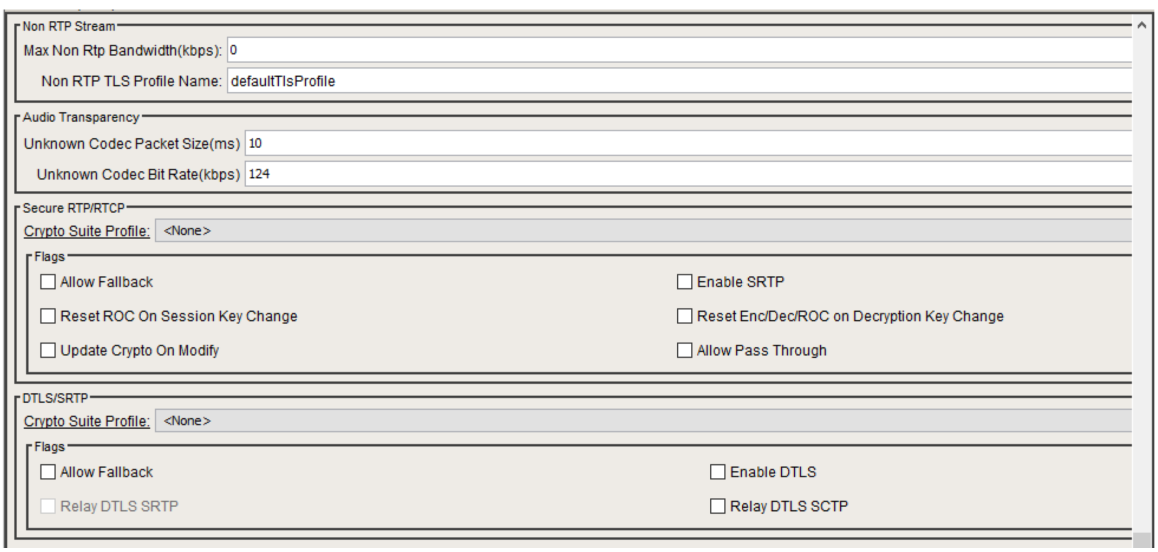
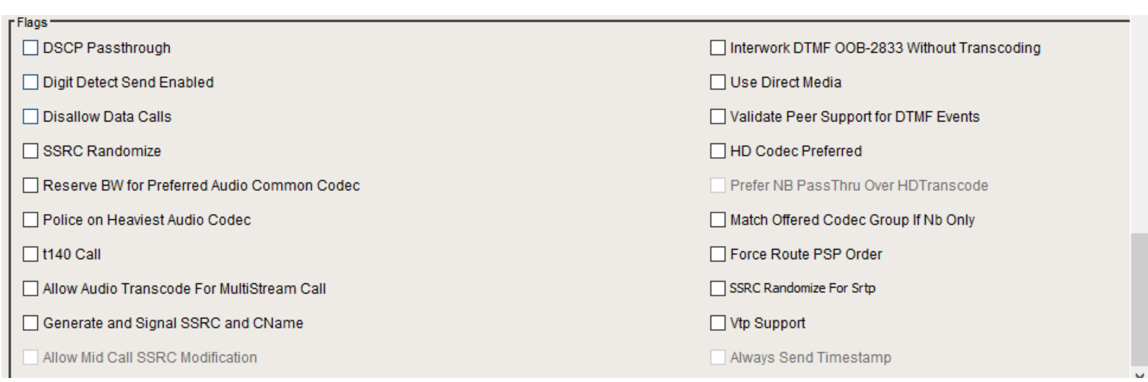


Figure37:



Packet Service Profile OUT

Figure38:

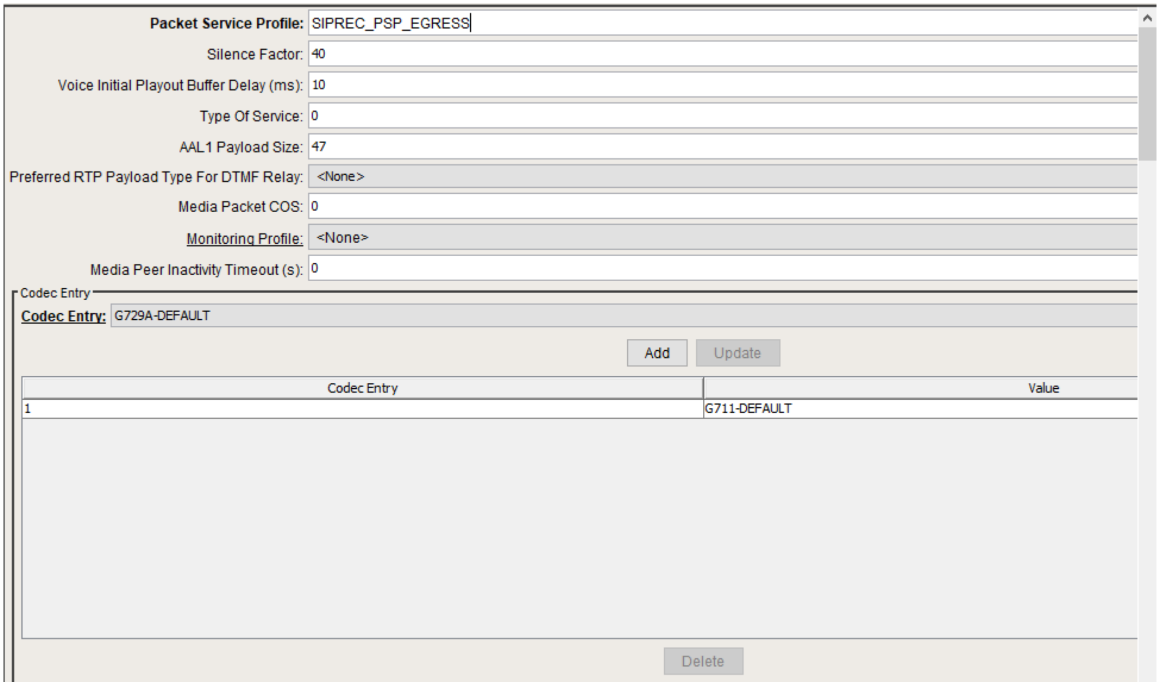


Figure39:

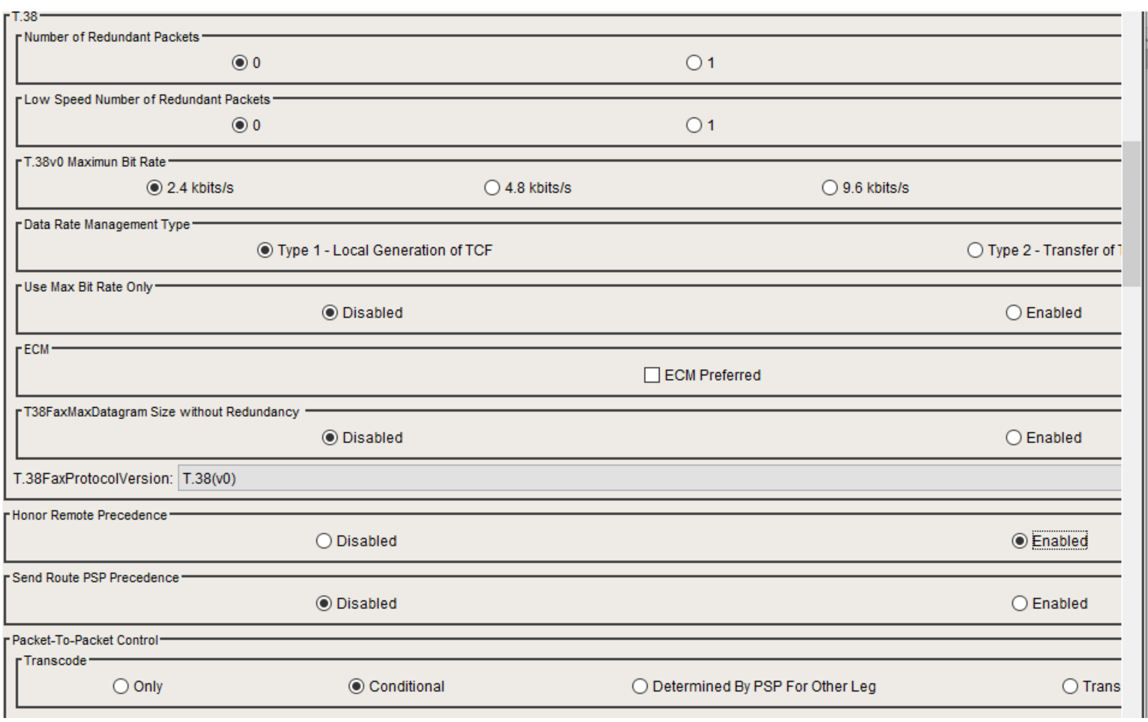


Figure40:

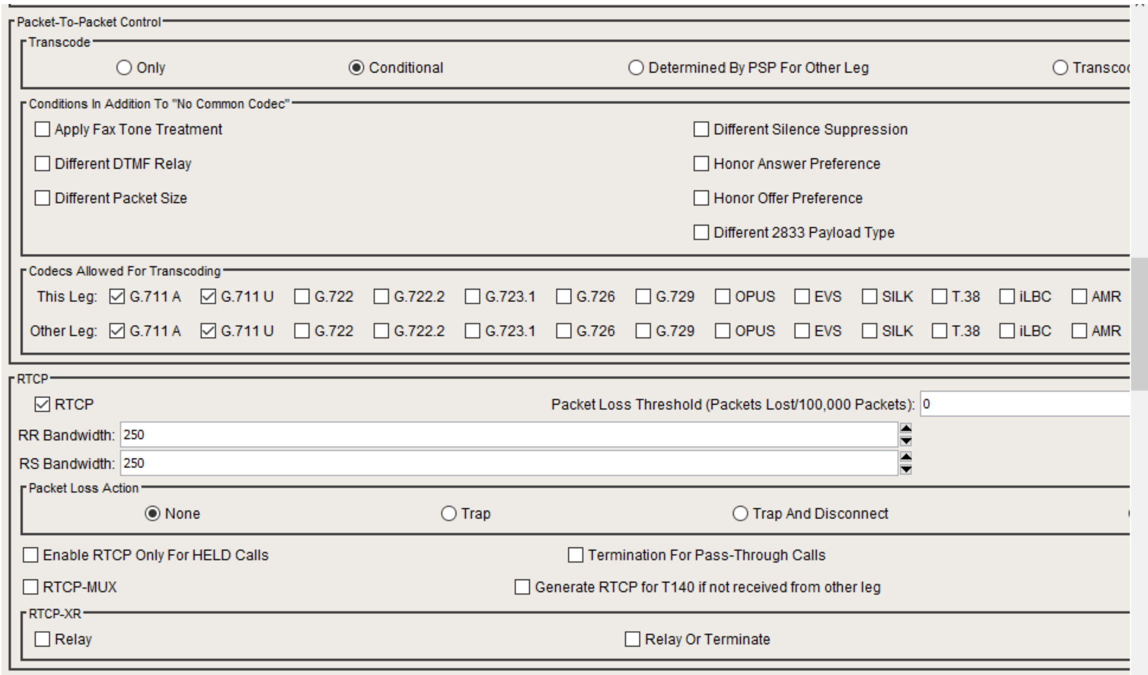


Figure41:

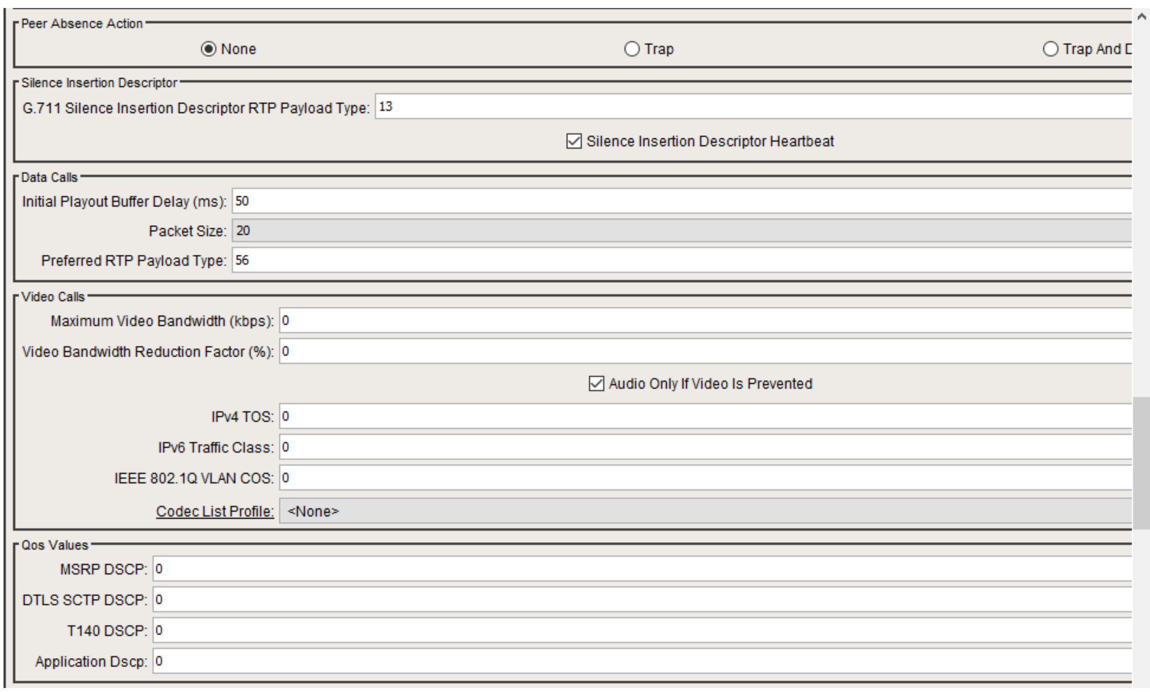


Figure 42:

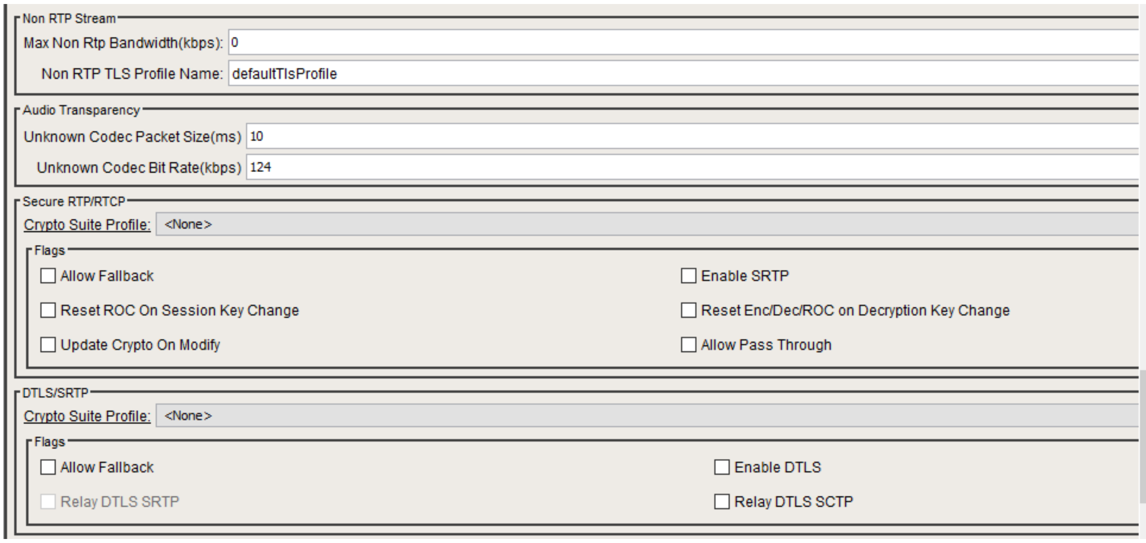
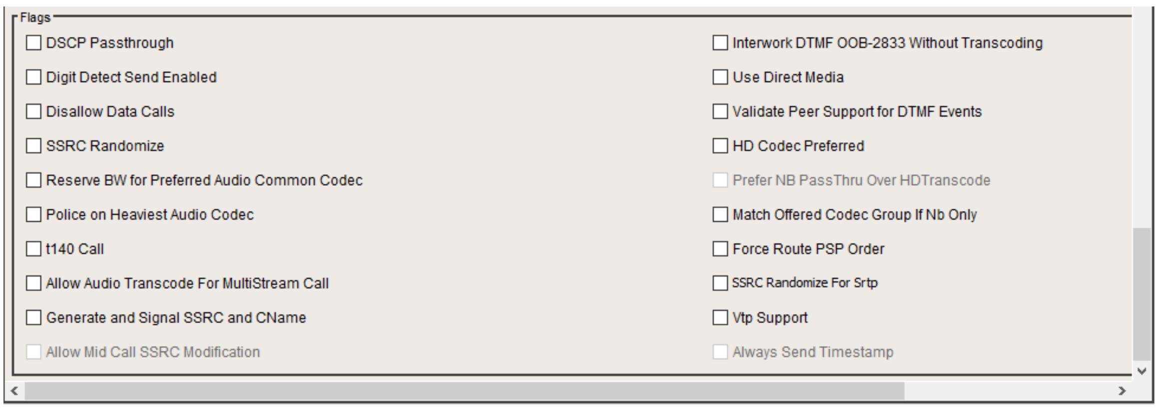


Figure 43:



Configuring IP Signaling Peer Group

IP Peer is an entity of the Session Border Controller, which is configured inside the Zone. It acts as a destination endpoint for the call to be routed towards. An IP Peer constitutes an IPv4/IPv6 address or a Fully Qualified Domain Name (FQDN) with a port number.

Figure 45:

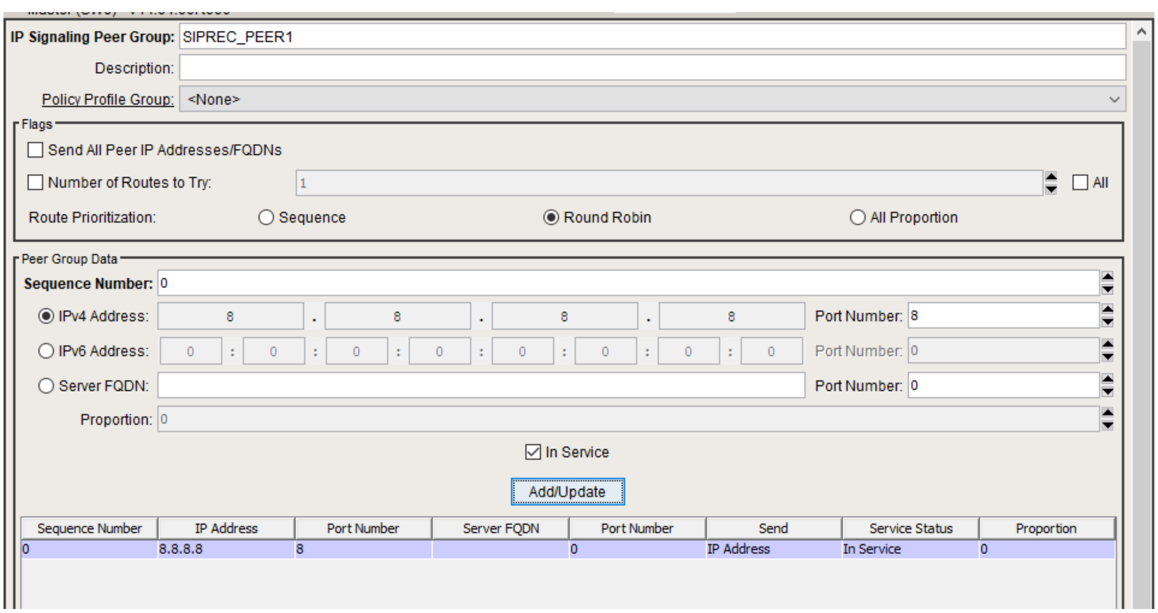
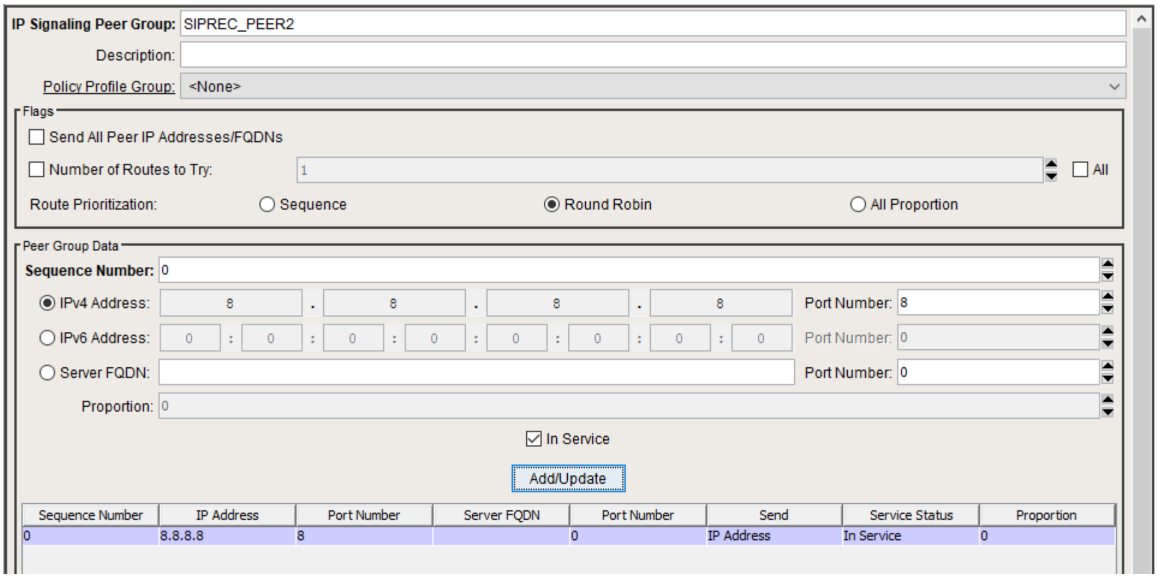


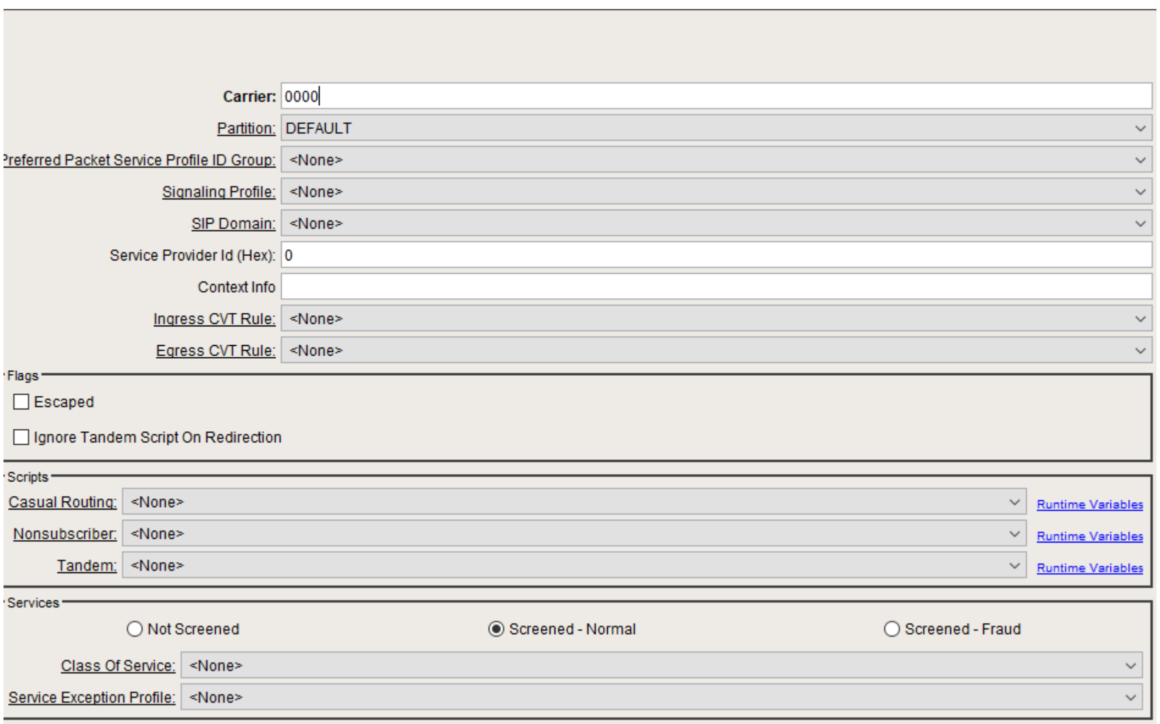
Figure 46:



Configuring Carrier

Please note that we have used default Carrier '**0000**' for our testing.

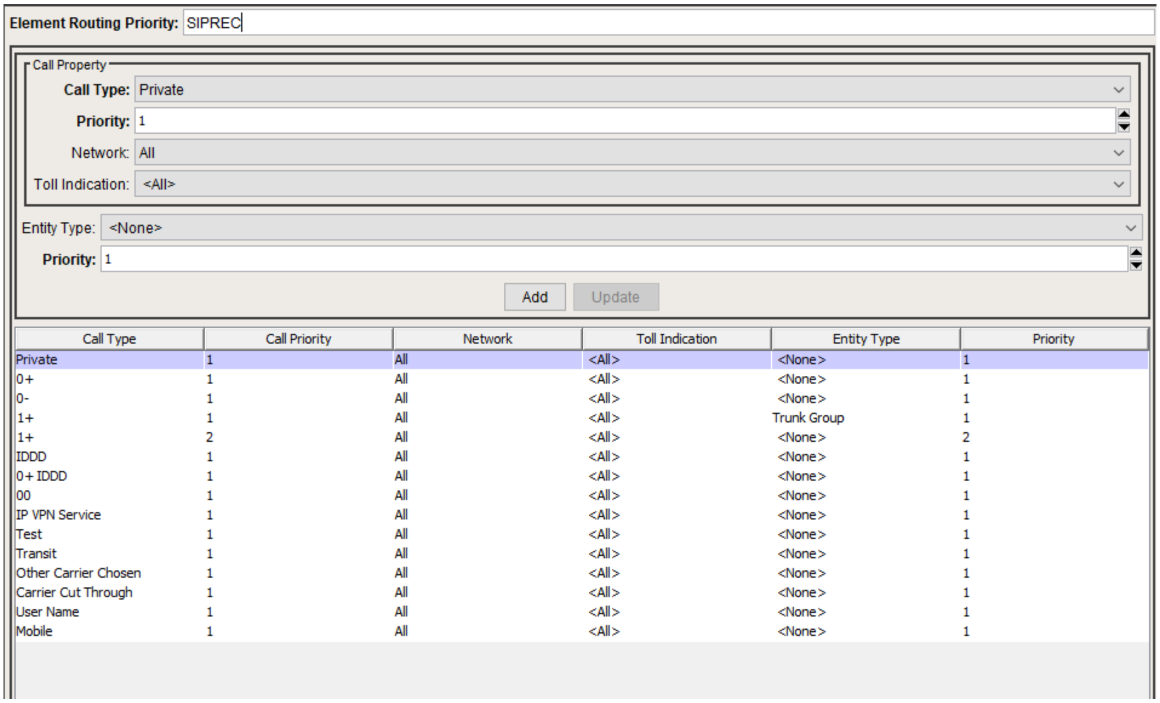
Figure 47:



Configuring Element Routing Priority Profile

Please note that we have cloned and used default Element Routing Priority for our testing.

Figure 48:



Configuring SignalingProfile

Please note that we have used default Signaling Profile **'DEFAULT_IP_PROFILE'** for our testing.

Figure 49:

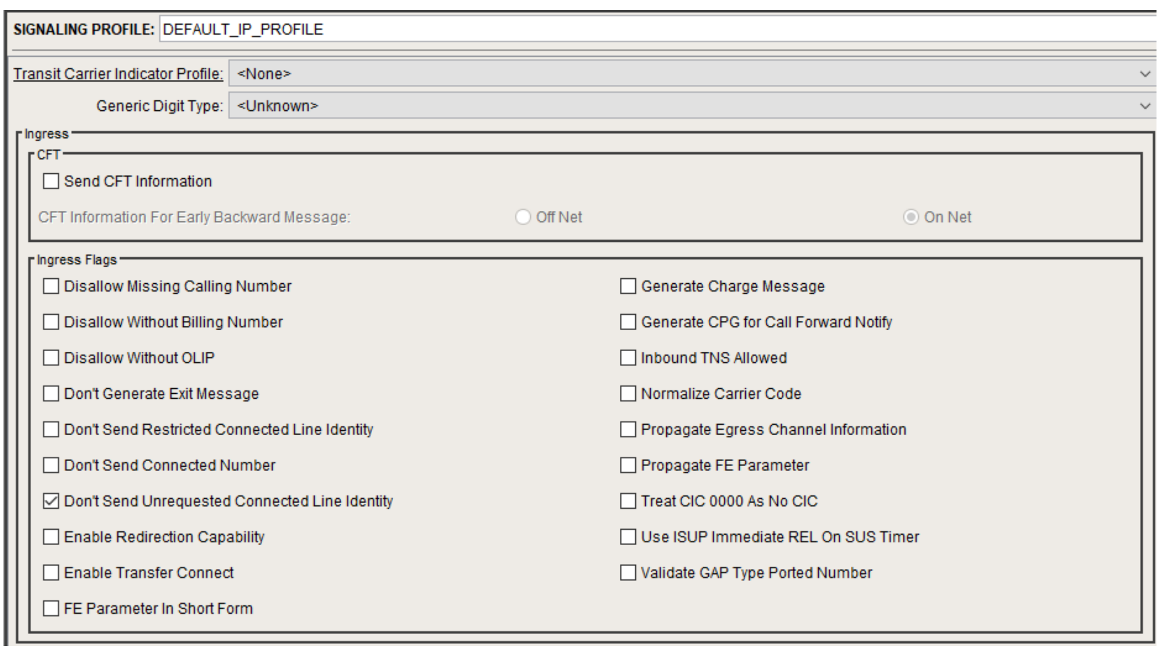


Figure 50:

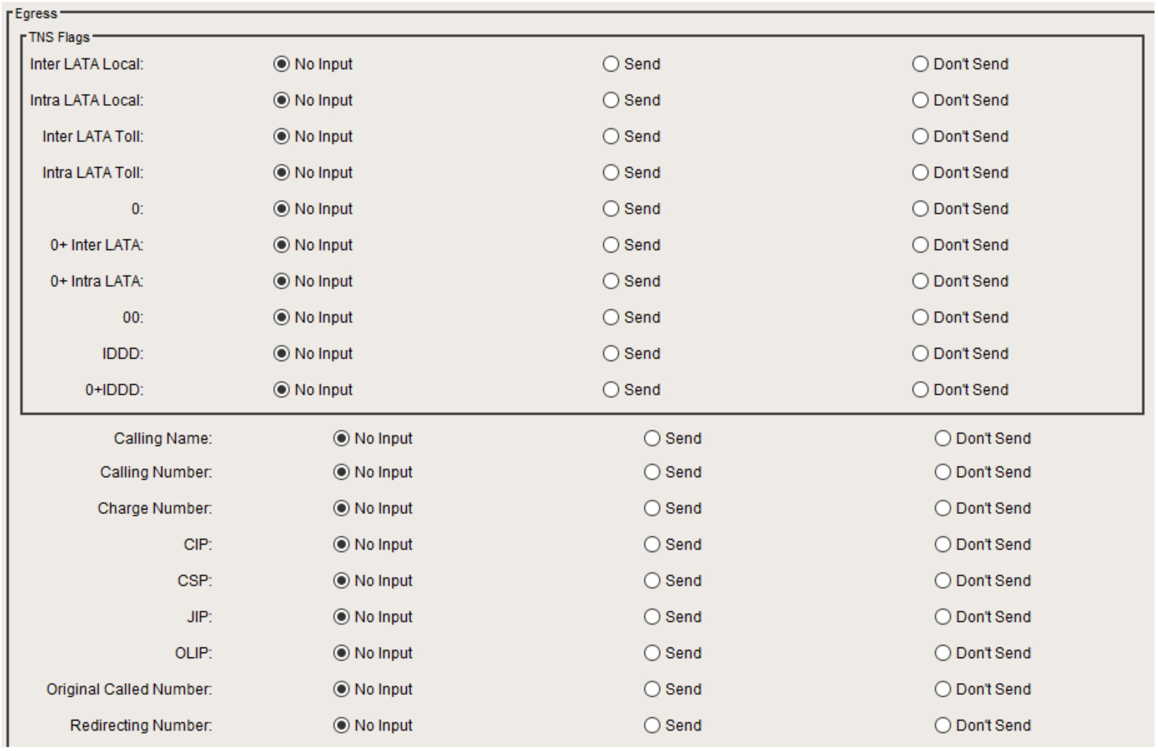


Figure 51:

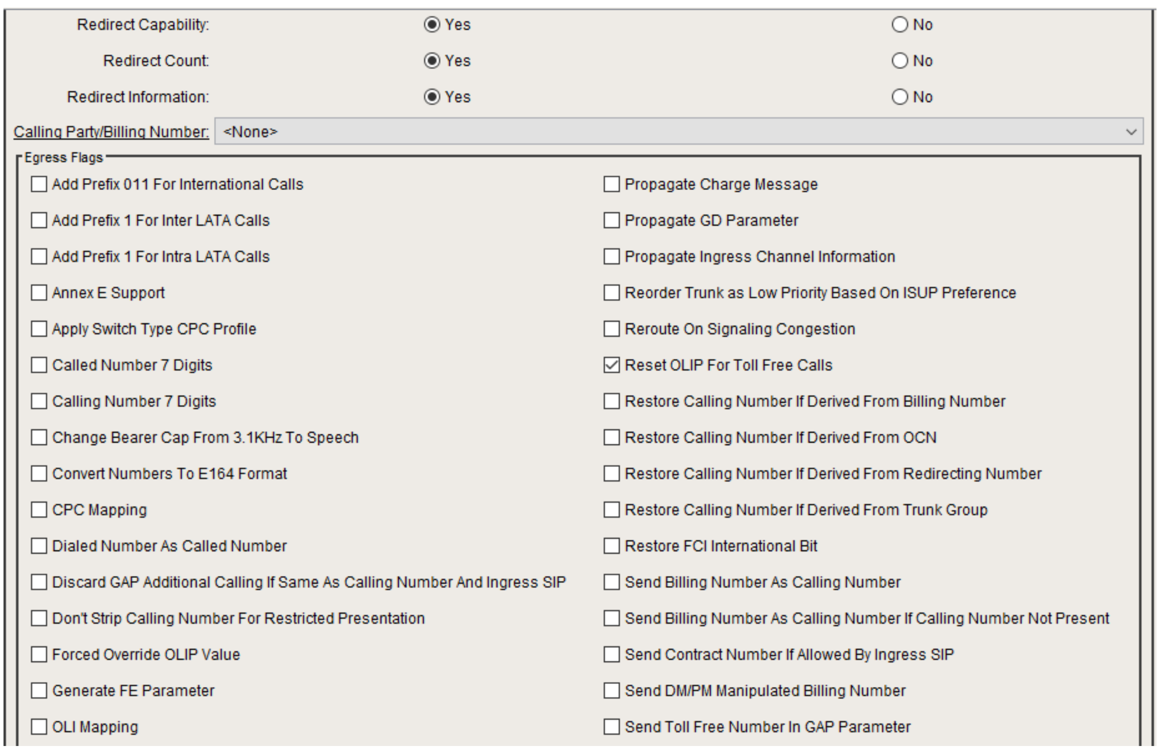


Figure 52:

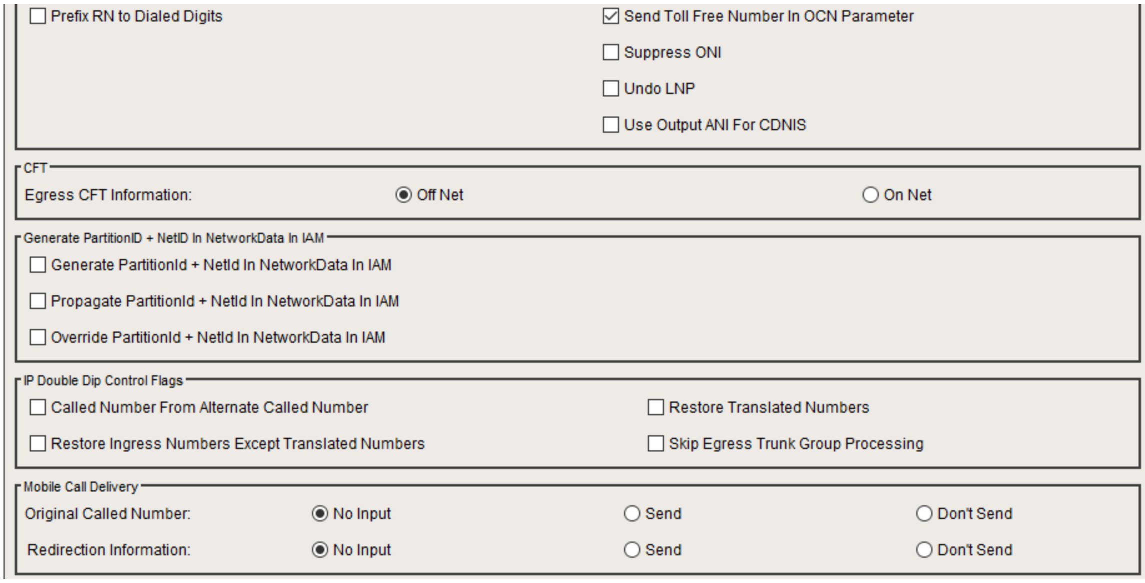
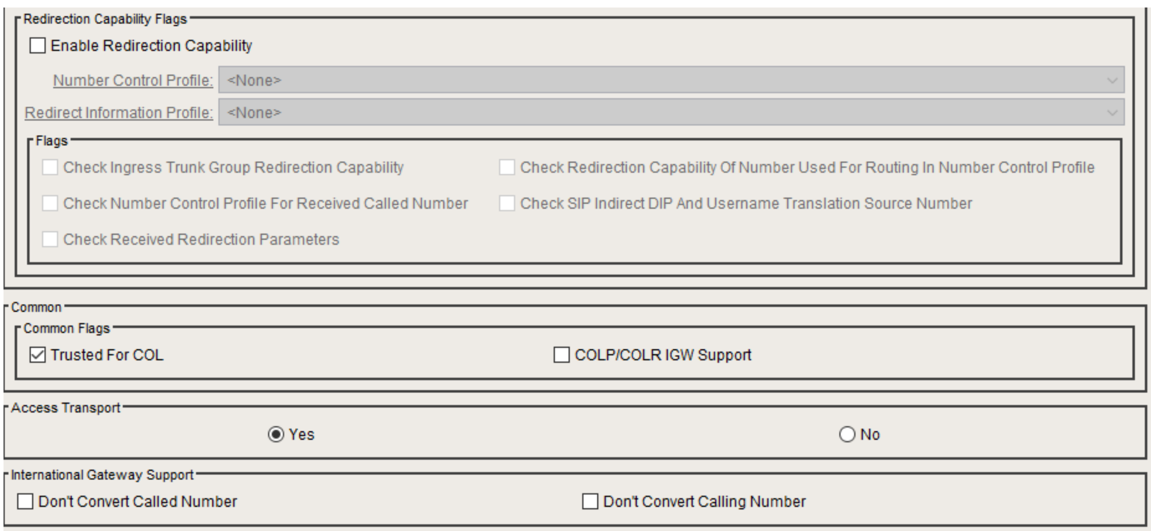


Figure 53:



Configuring Feature Control Profile

Please note that we have used default Feature Control Profile 'DEFAULT_IP' for our testing.

Figure 54:

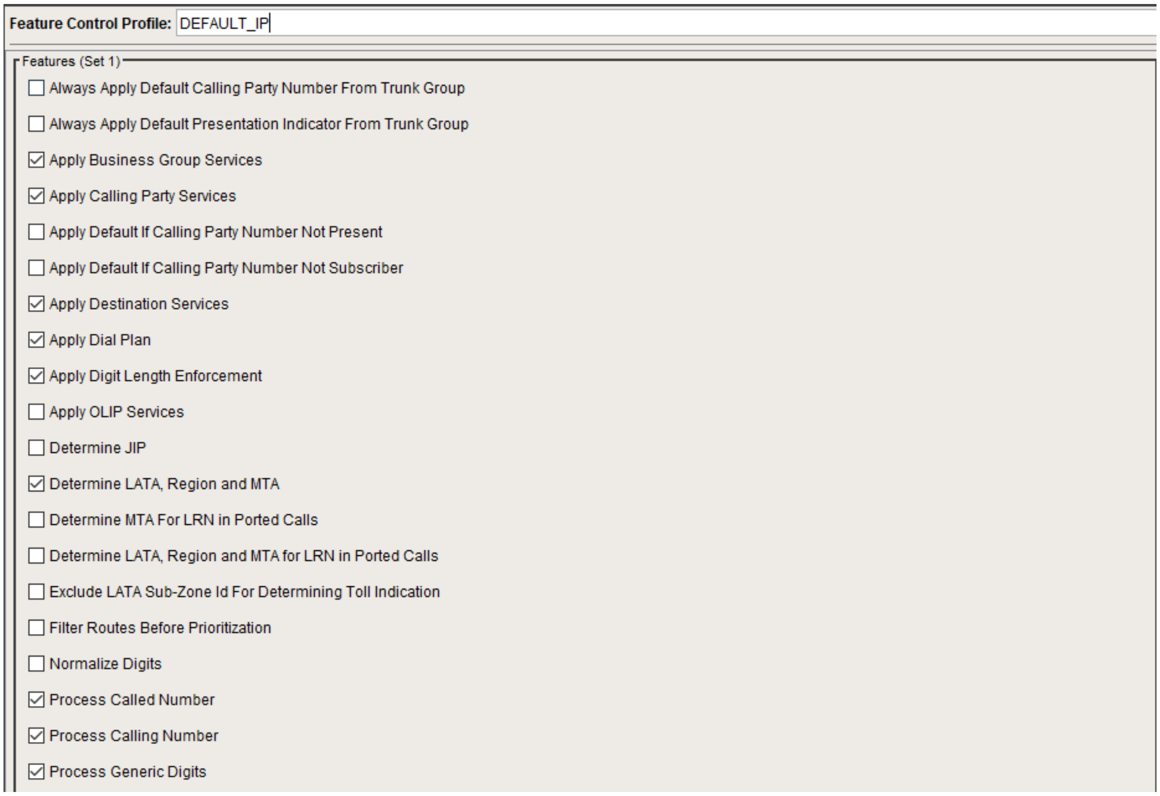


Figure 55:

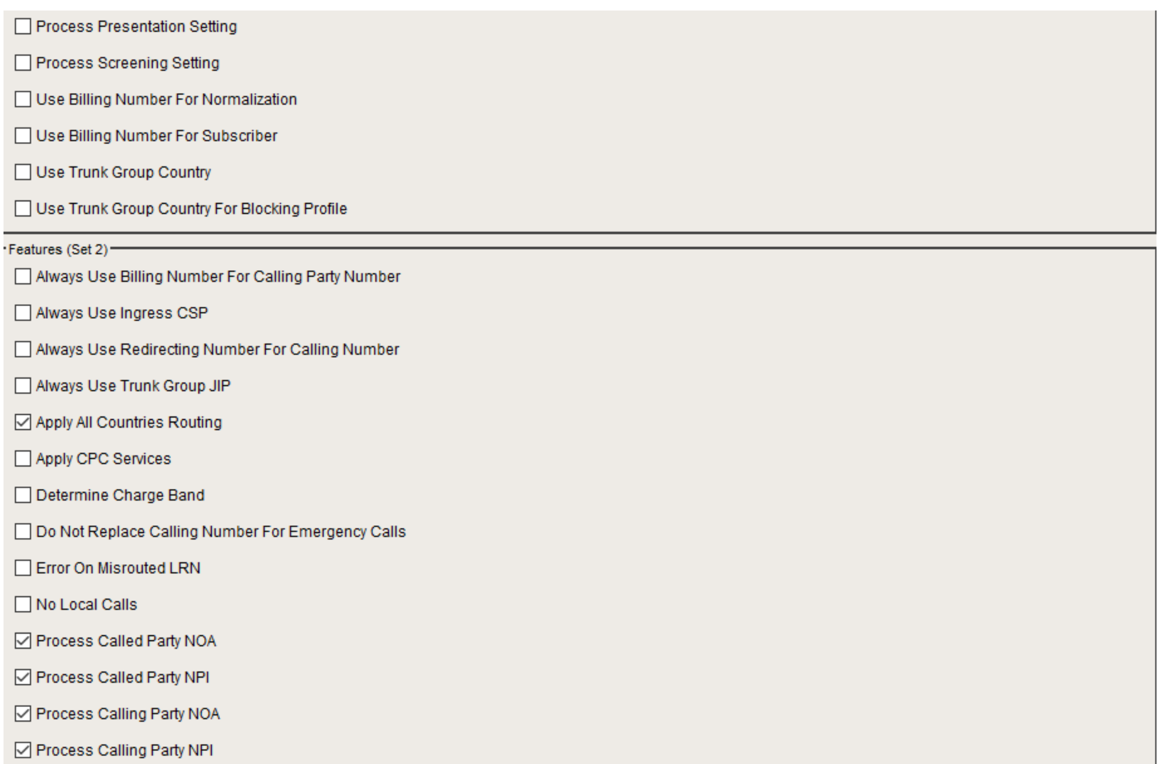


Figure 56:

Figure 57:

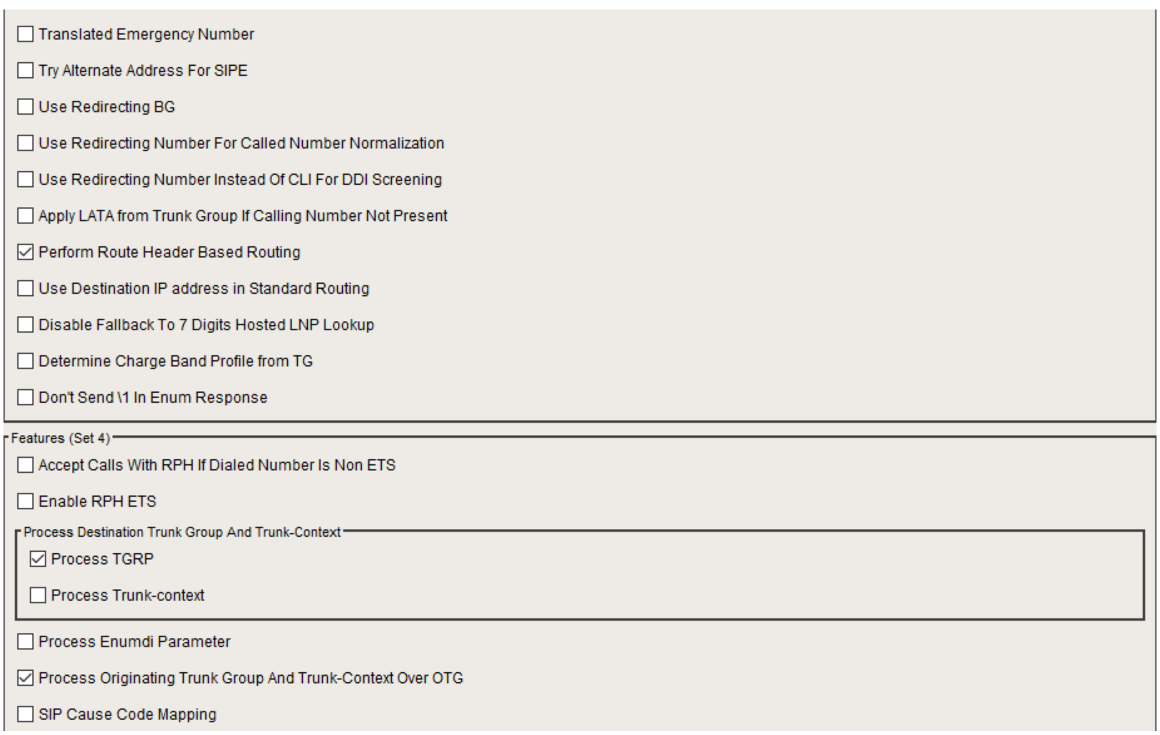


Figure 58:

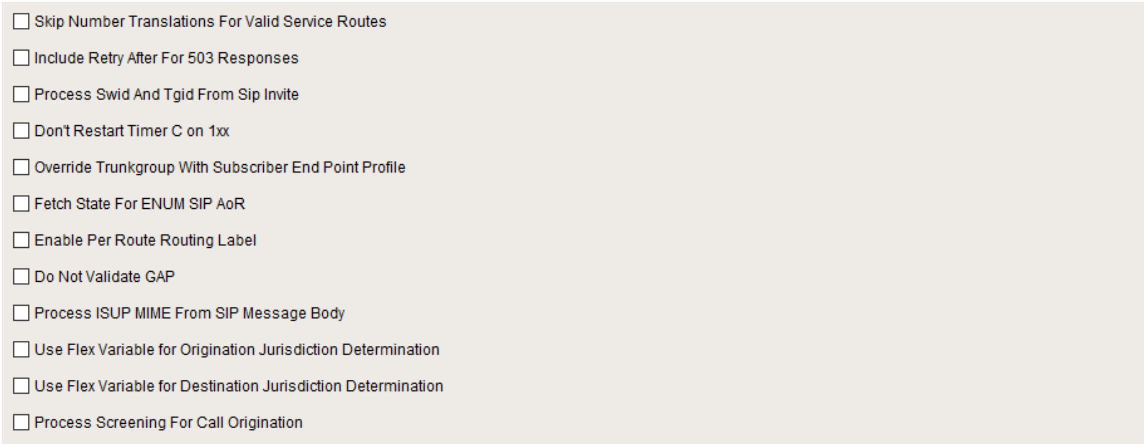
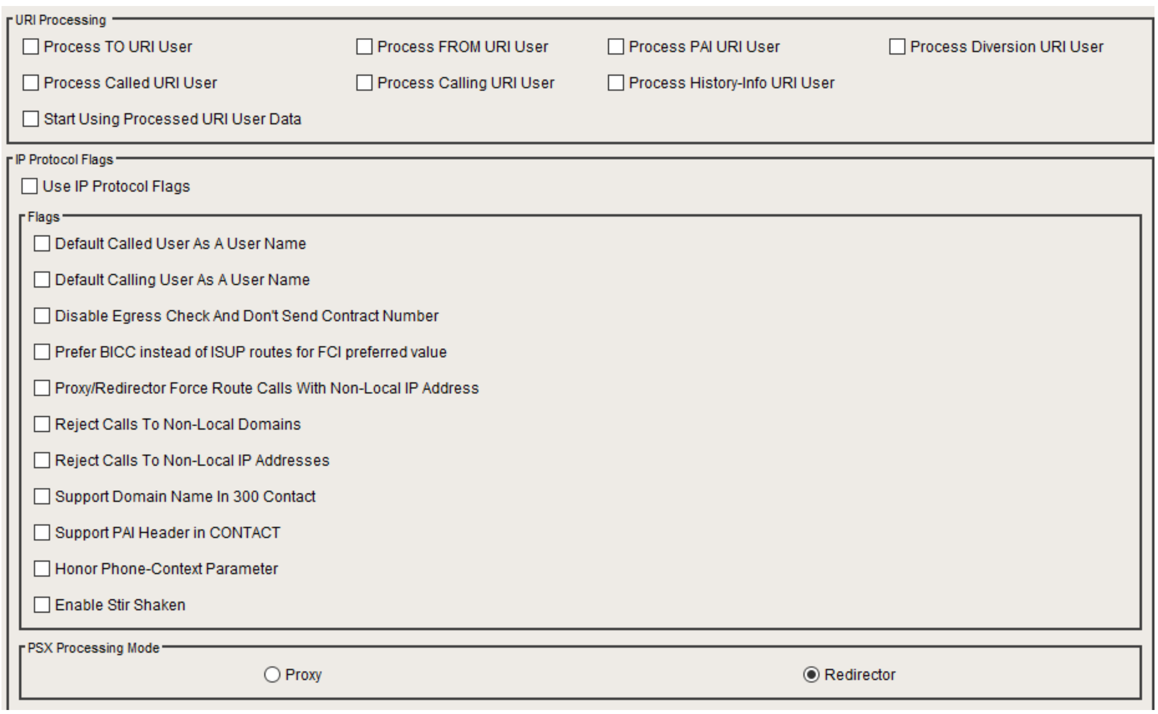


Figure 59:



Configuring Trunk Groups

Create two Trunk Groups for Ingress and Egress and associate the Trunk Groups to the gateway created in Step-1.

> **Warning**
> Mandatory! You must capitalize SIP Trunk Group names.

Trunk Group IN

Follow the instructions below for Ingress Trunk Group.

Figure 60:

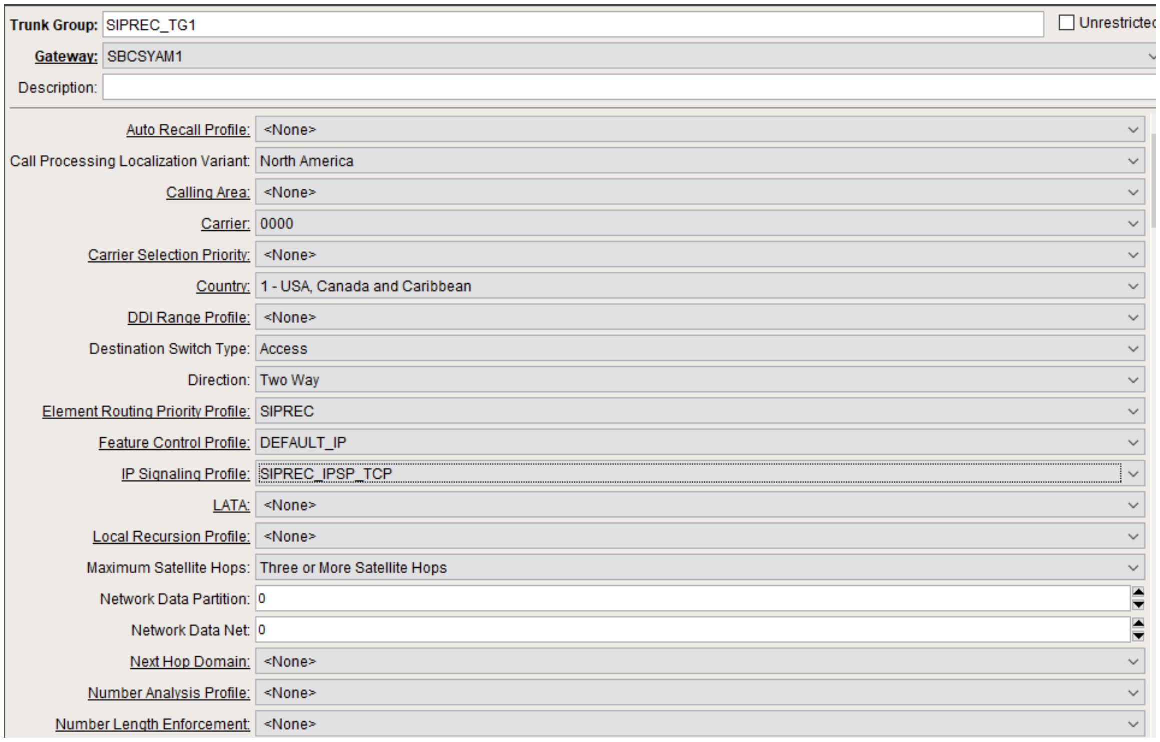


Figure 61:

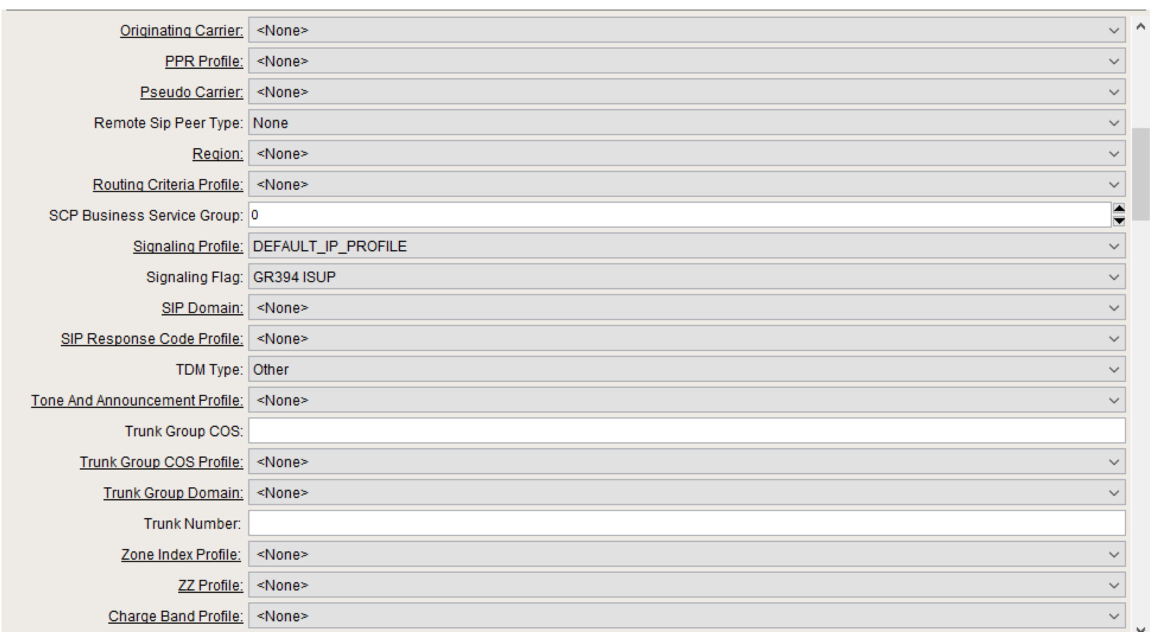


Figure 62:

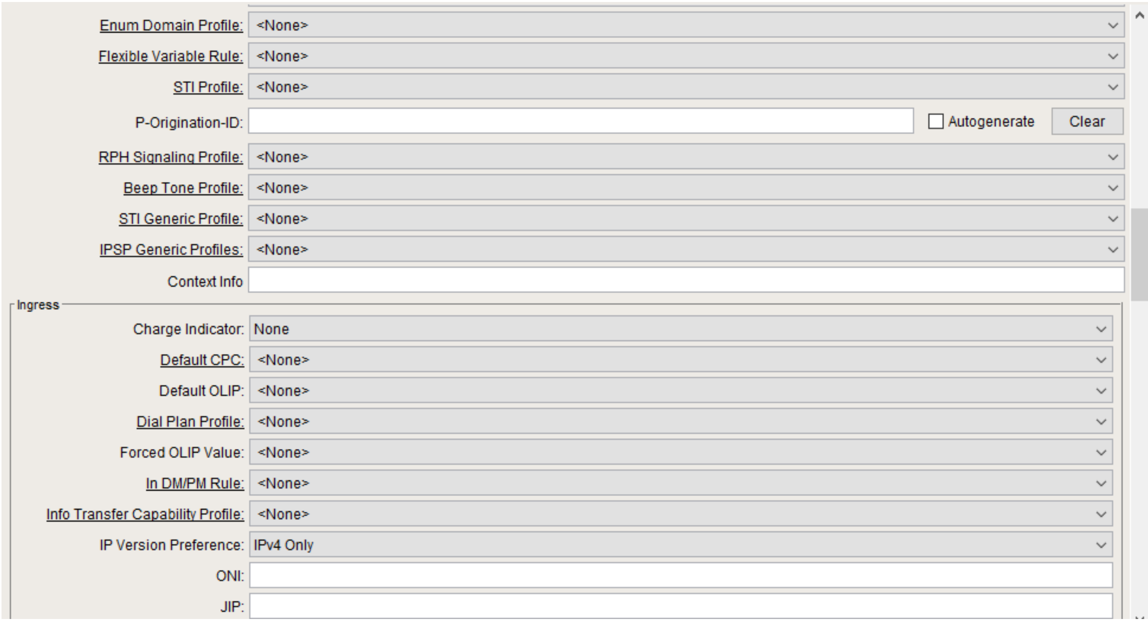


Figure 63:

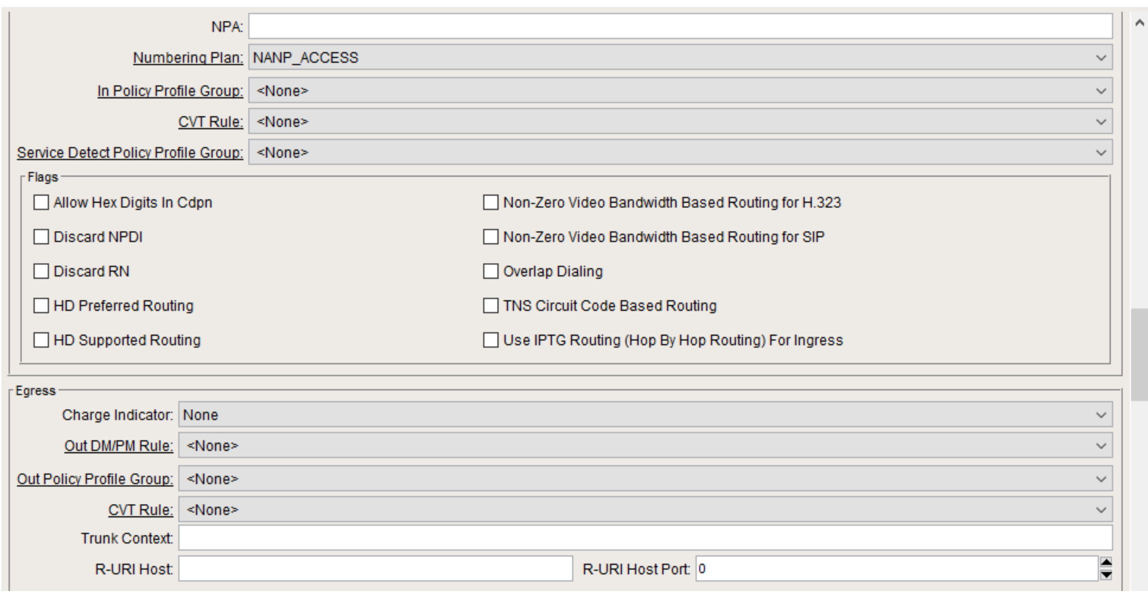


Figure 64:

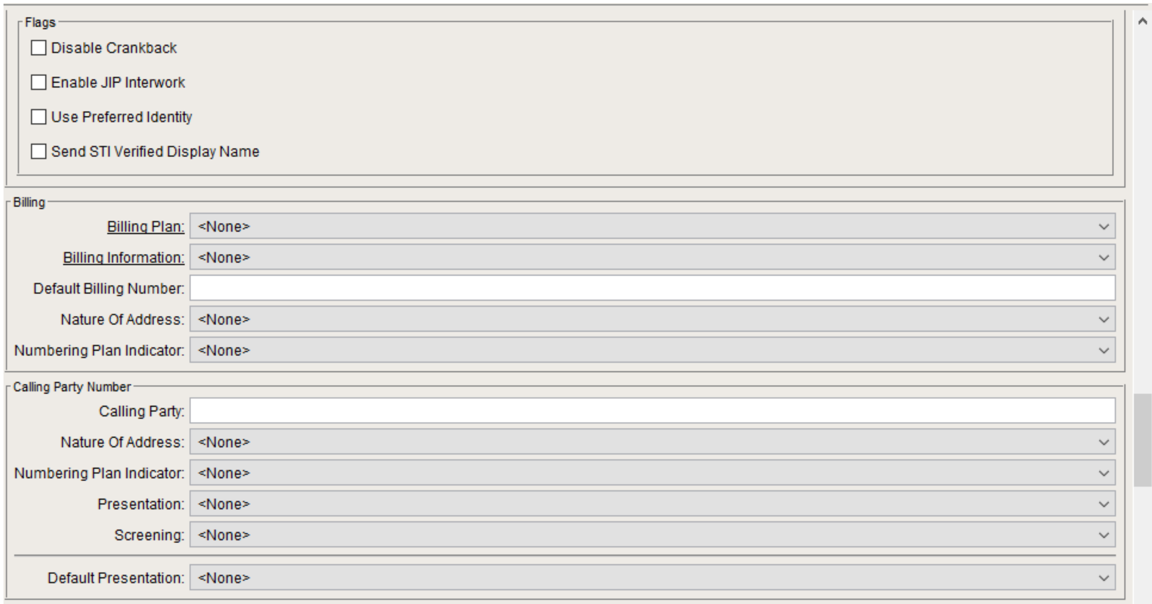


Figure 65:

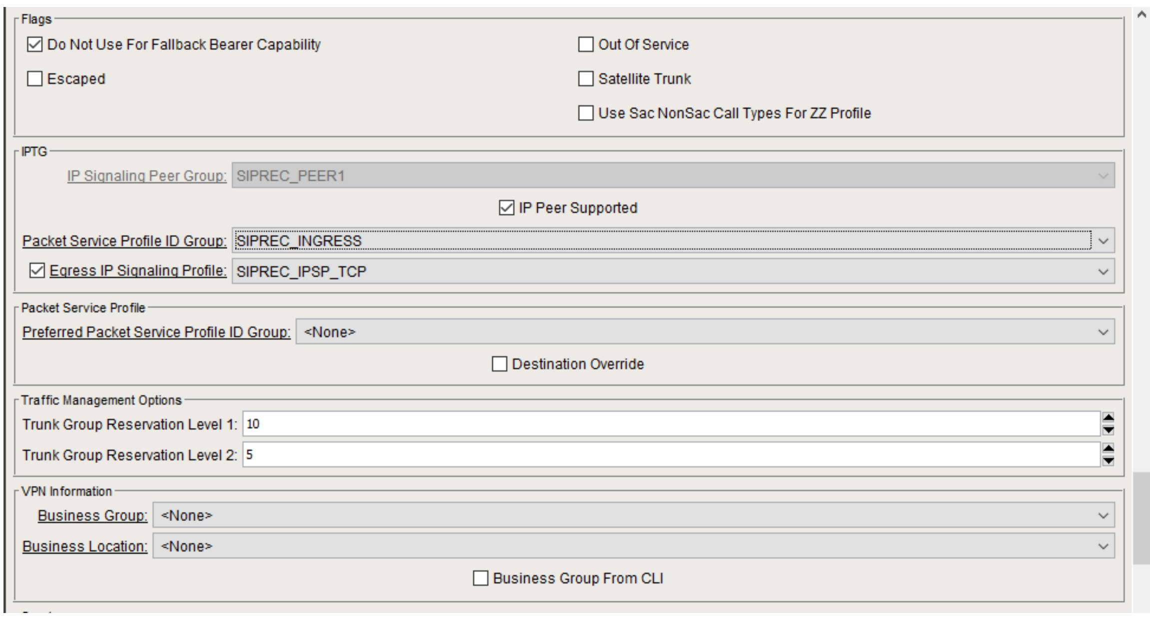
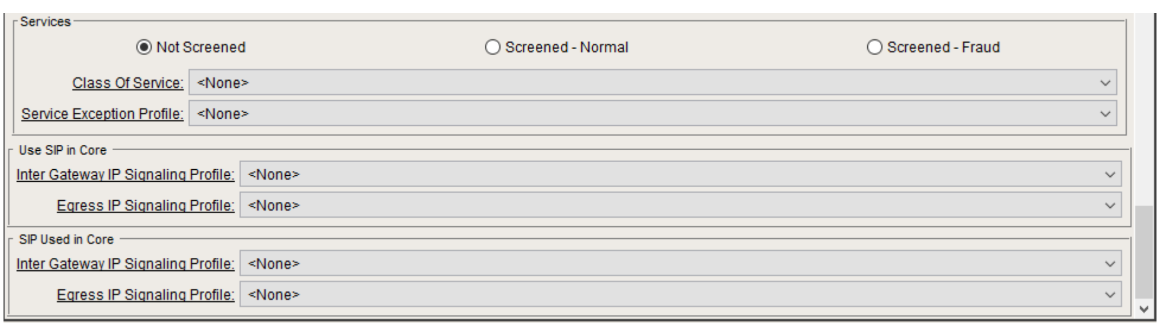


Figure 66:



Trunk Group OUT

Follow the instructions below for Egress Trunk Group.

Figure 67:

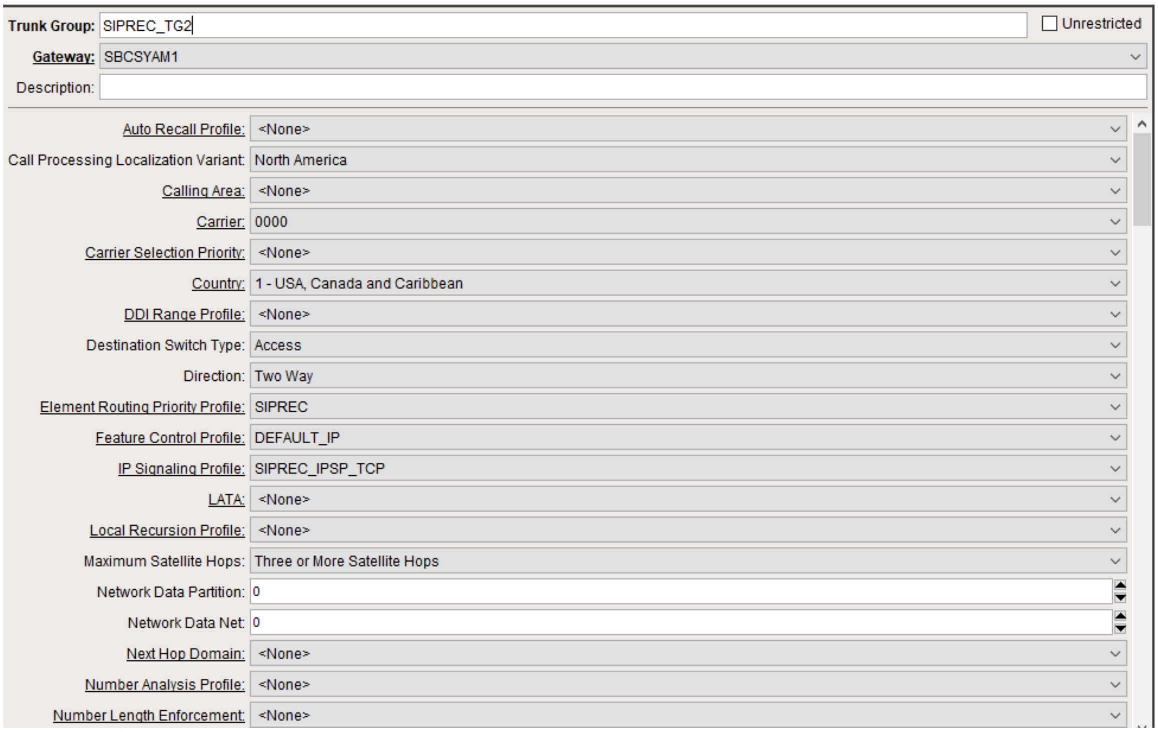


Figure 68:

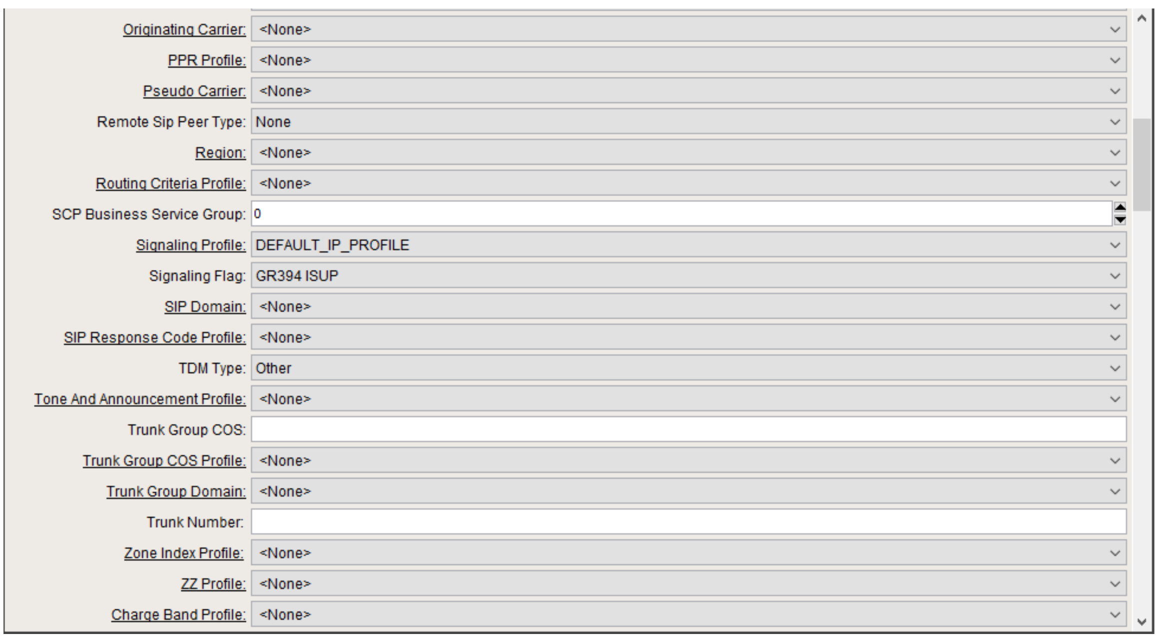


Figure 69:

Figure 70:

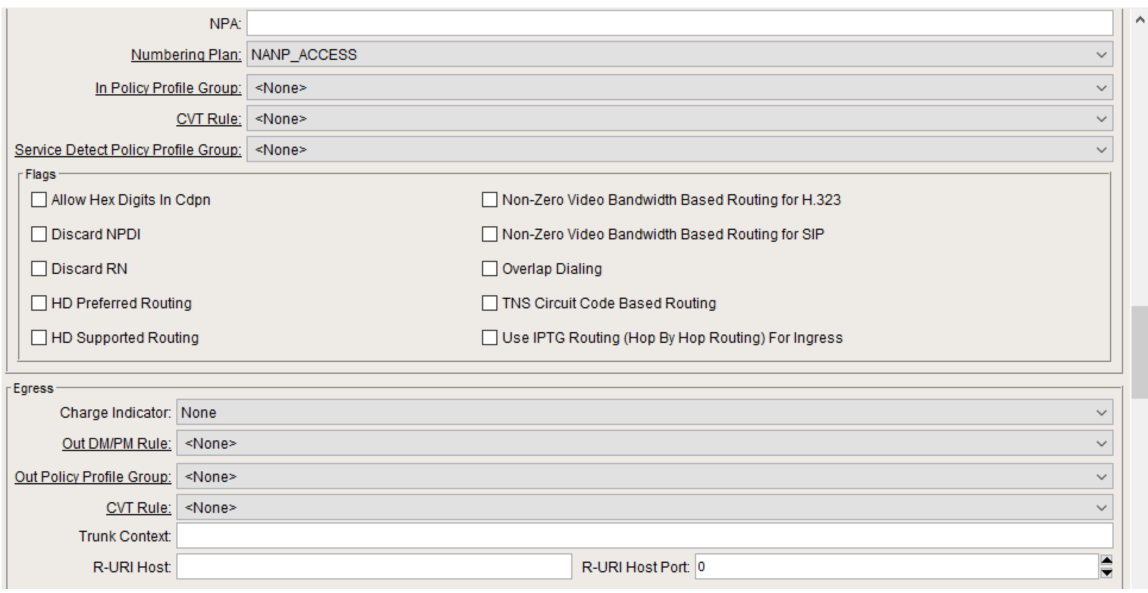


Figure 71:

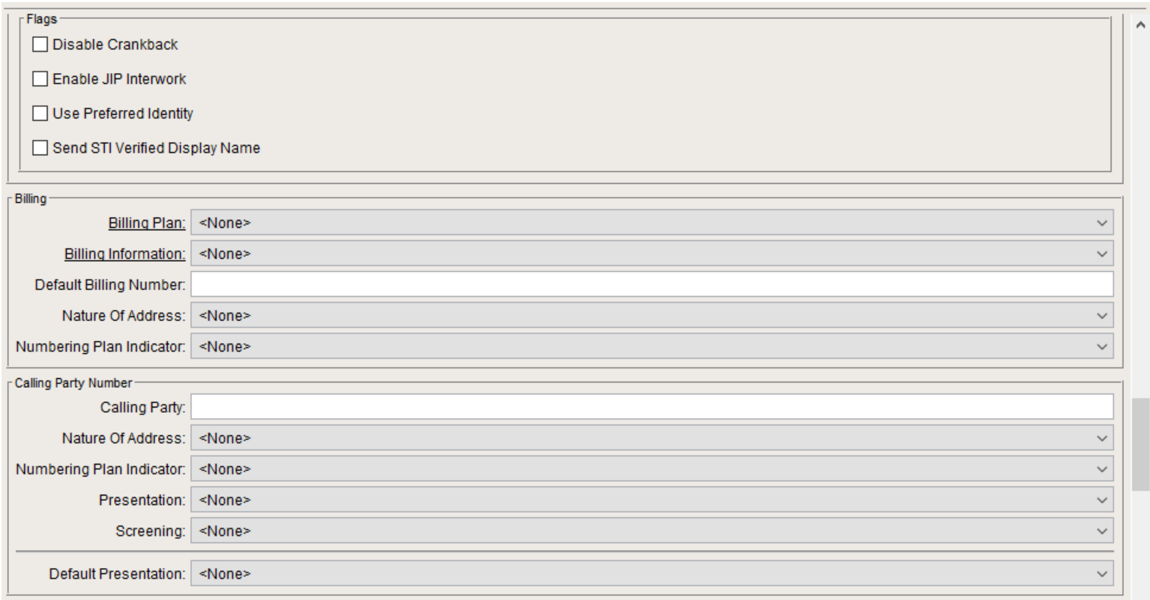


Figure 72:

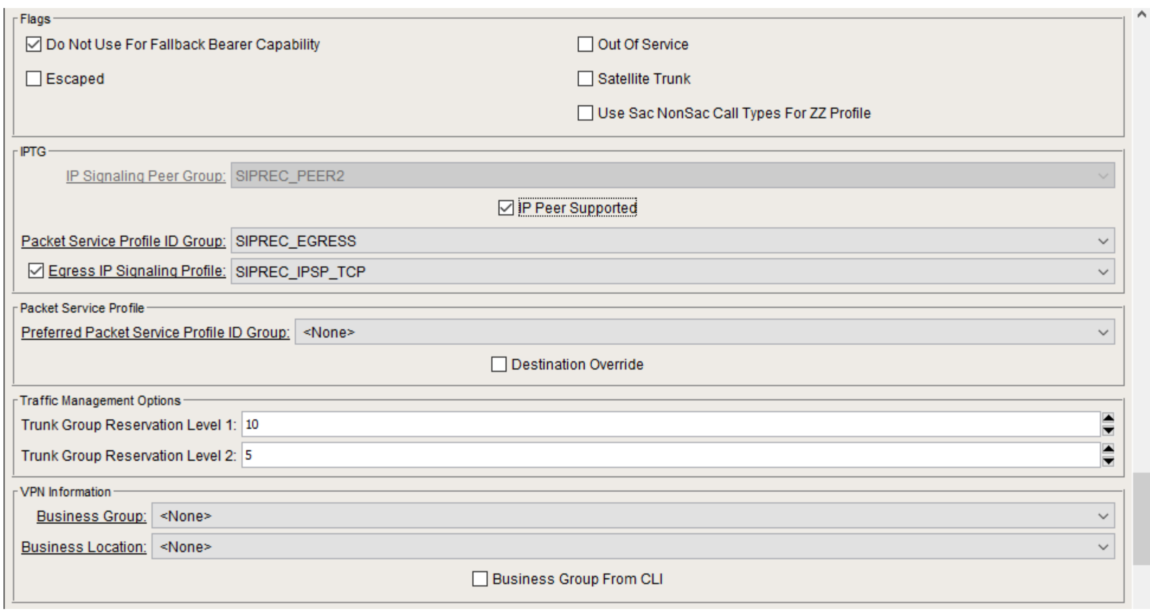
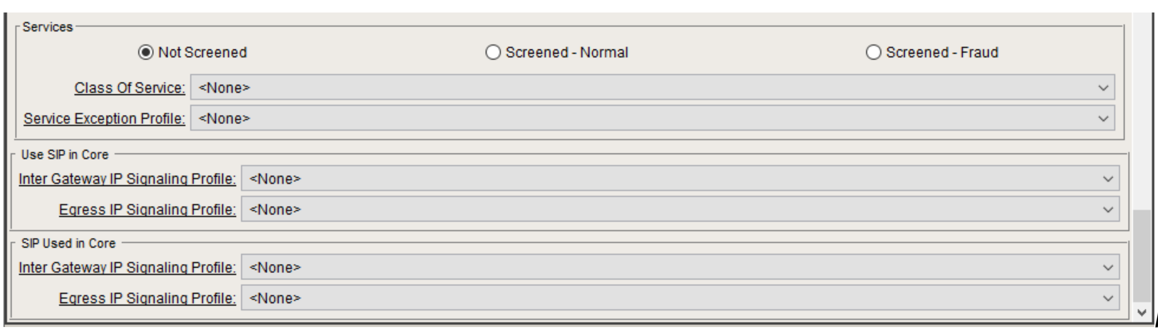


Figure 73:



Configuring Routes

Routing allows you to send calls to the correct destination. You can use routing options based on your requirements. Configure the standard and specific routes (with usernames) to ensure that no matter how the called party is addressed (a number or username), the SBC routes the message to the Core.Create Route entries for standard Trunk Group routing with Matching Criteria and a Routing Label destination.

Routing Label

A routing label is associated with a route.Each route includes a gateway/trunk group pair. Routing labels provide the link between an entry in the Standard Route table and the set of routes associated with that Standard Route table entry.

Routing Label 1

Figure 74:

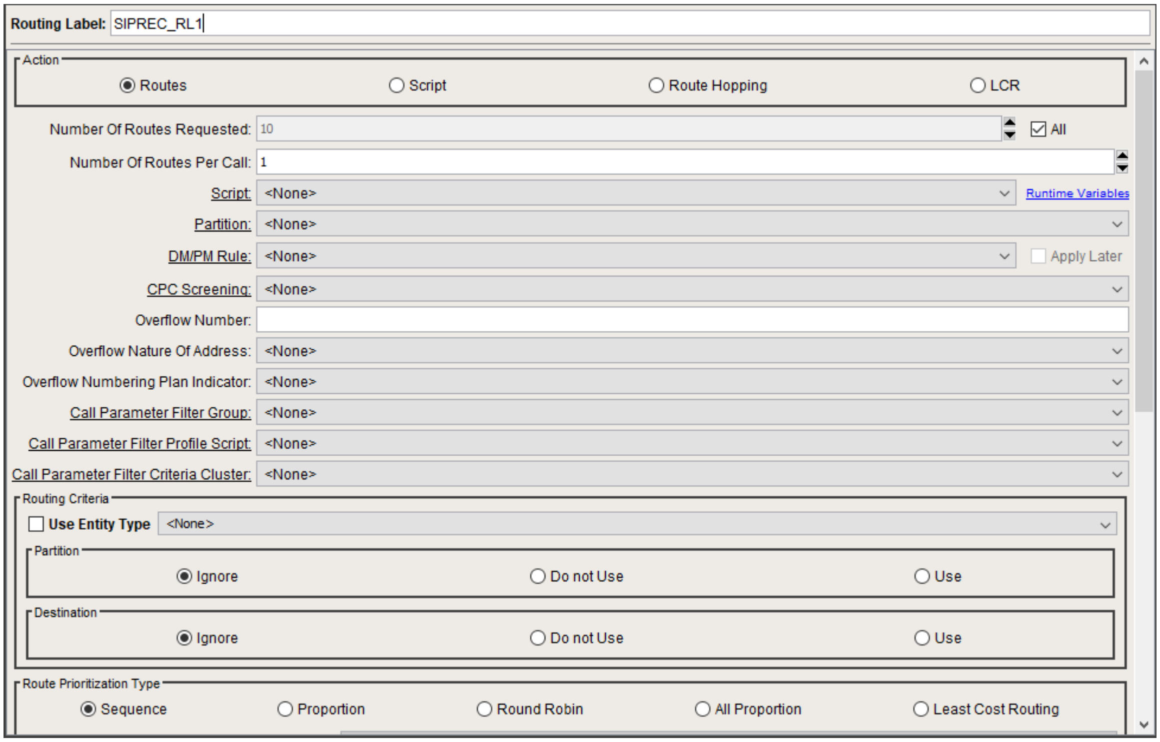


Figure 75:

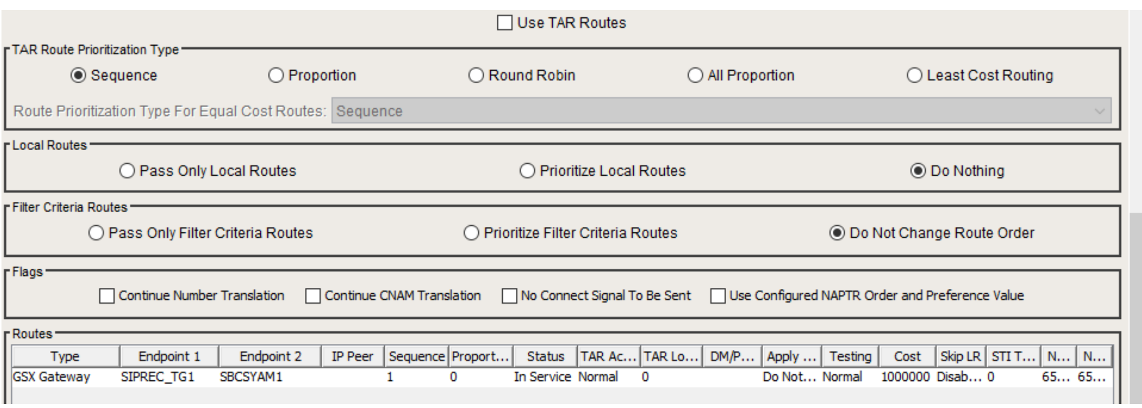
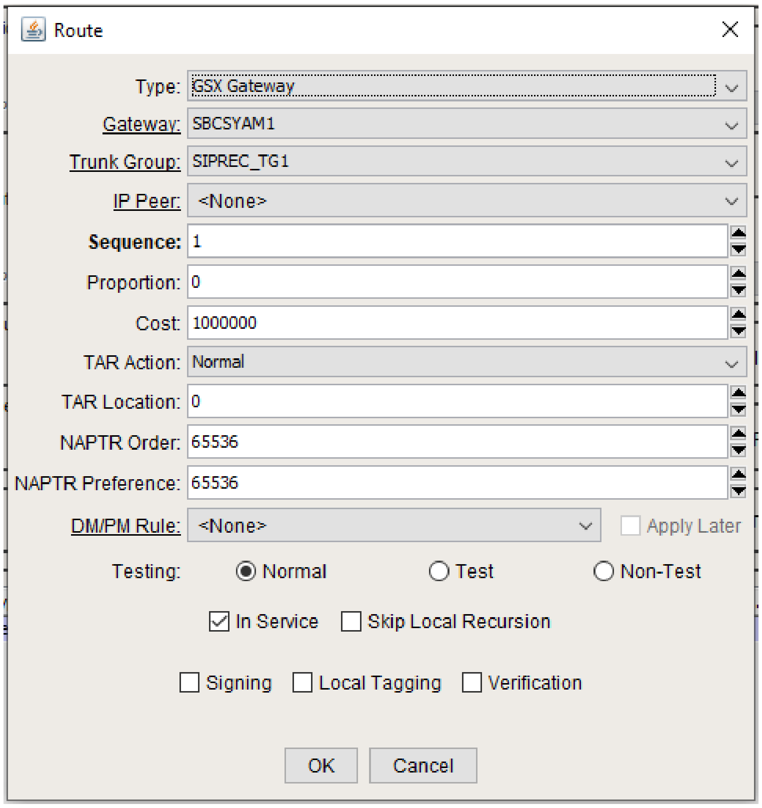


Figure 76:



Routing Label 2

Figure 77:

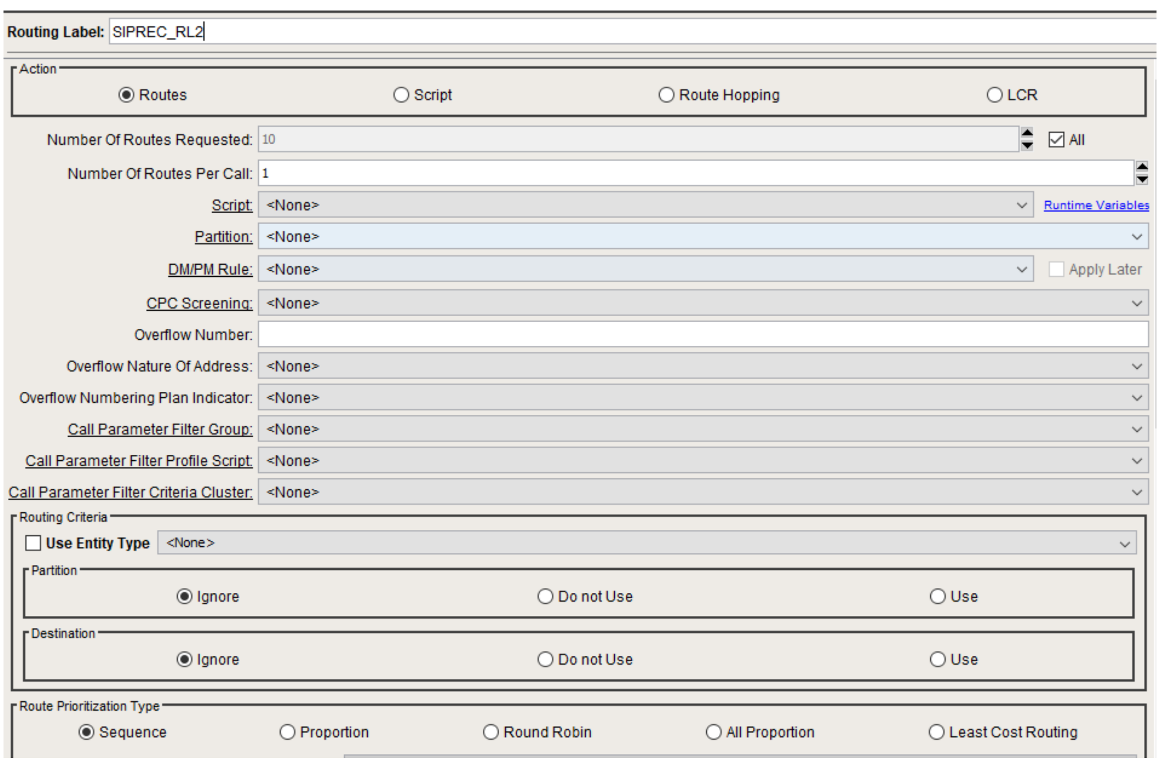


Figure 78:

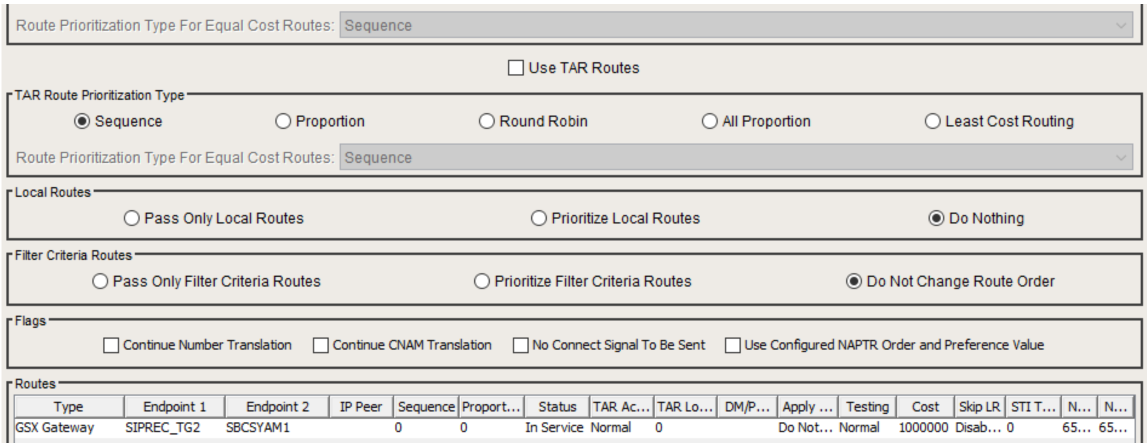
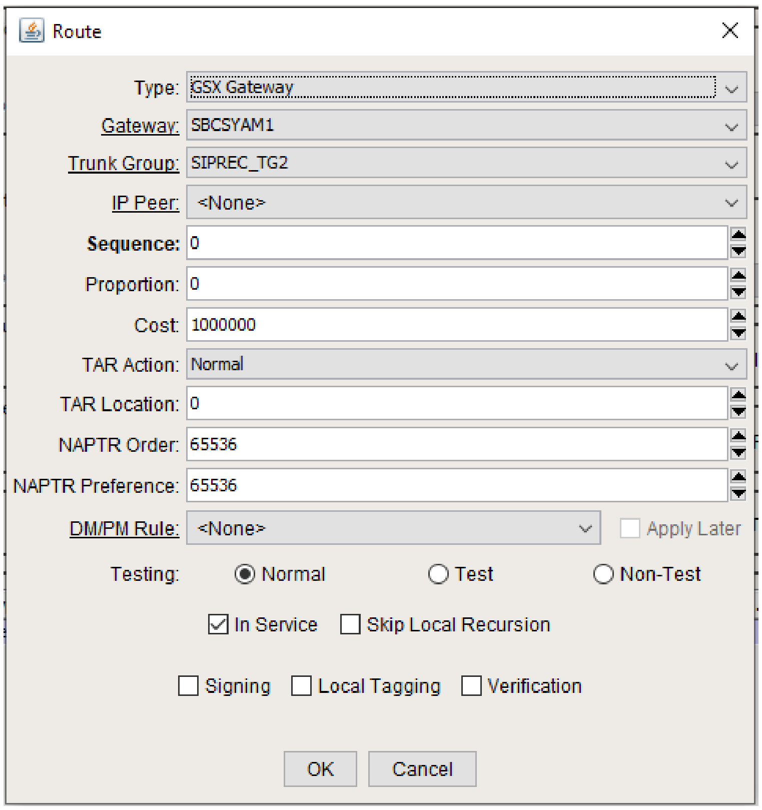


Figure 79:

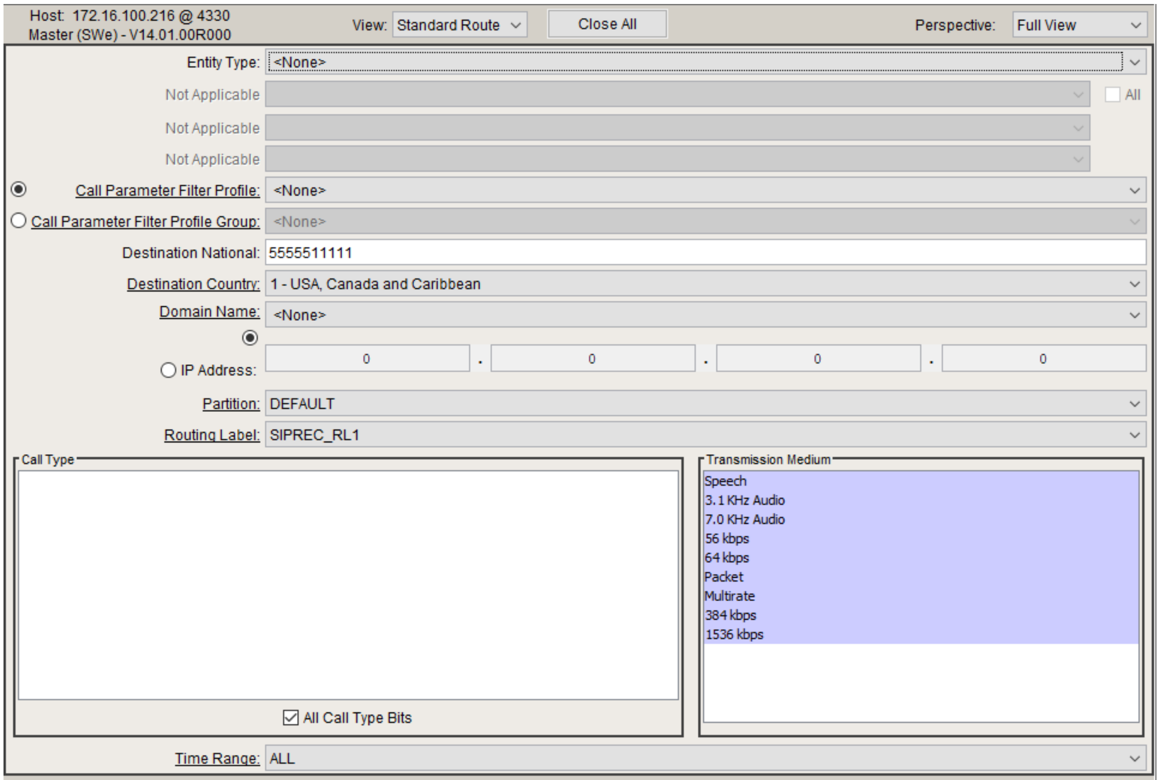


Routes

Routing allows you to send calls to the correct destination. You can use routing options based on your requirements. Configure the standard and specific routes (with usernames) to ensure that no matter how the called party is addressed (a number or username), the SBC routes the message to the Core.Create Route entries for standard Trunk Group routing with Matching Criteria and a Routing Label destination.

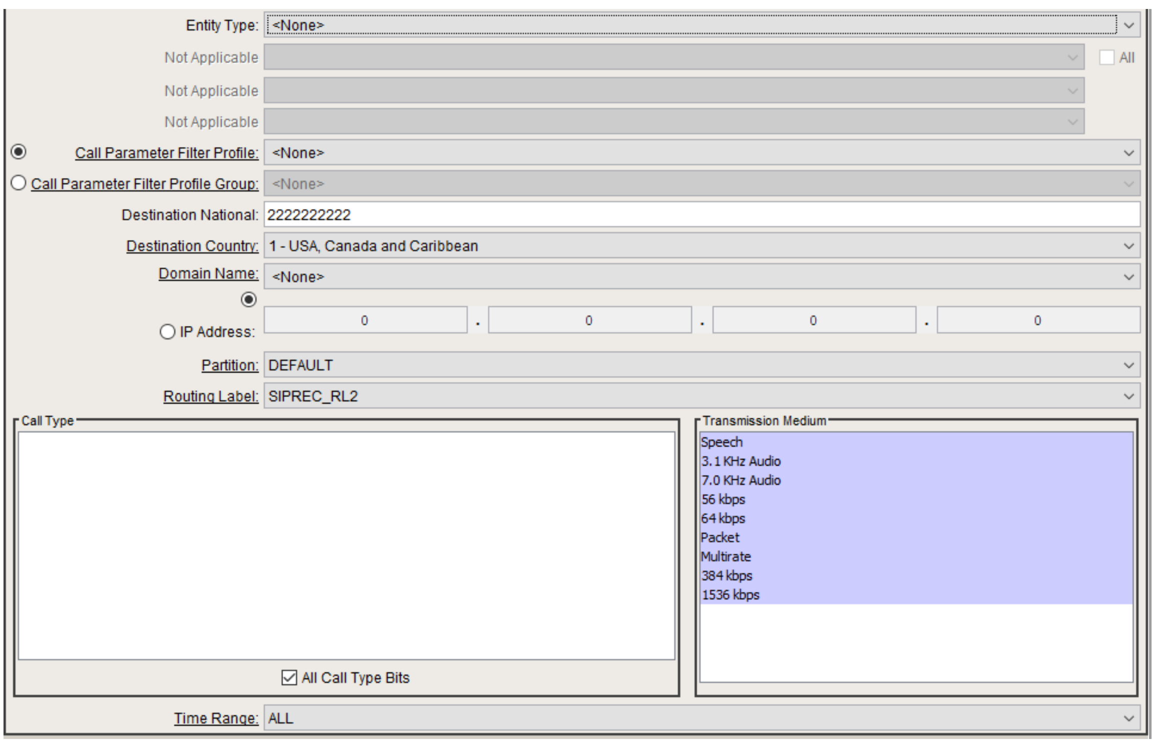
Route 1

Figure 80:



Route 2

Figure 81:



ConfiguringSIPRec

The PSX uses the following configurable objects when determining whether a call needs to be recorded or not:

- Recording Criteriacontain the rules to match for invoking call recording (this is the same for SIPREC and MCT).
- SRS Groupscontains multiple Recording profiles for SRS redundancy (up to 8).
  - Transport

- IP V4/V6 address port
    - Encryption data (for SRTP)
    - IP TG to be used by the SBC for RS session
    - Contains data of multiple SRS servers
- Recording Cluster profilecontains multiple SRS Groups for simultaneous recording (up to 4).

NICE Trunk Group

Create Trunk Group in PSX for SIPRec with the same name created above using SBC CLI. Duplicate default IP Signaling Profile and Packet Service Profile and map it to NICE TG.

Figure 82:

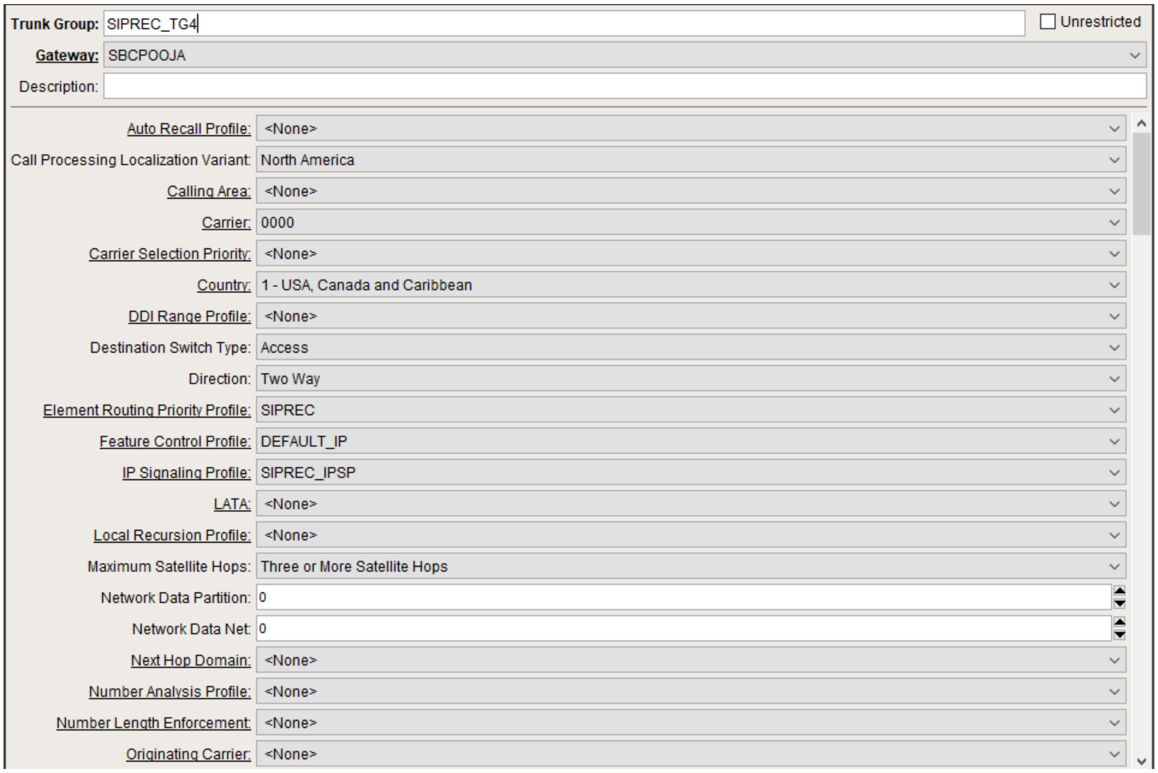


Figure 83:

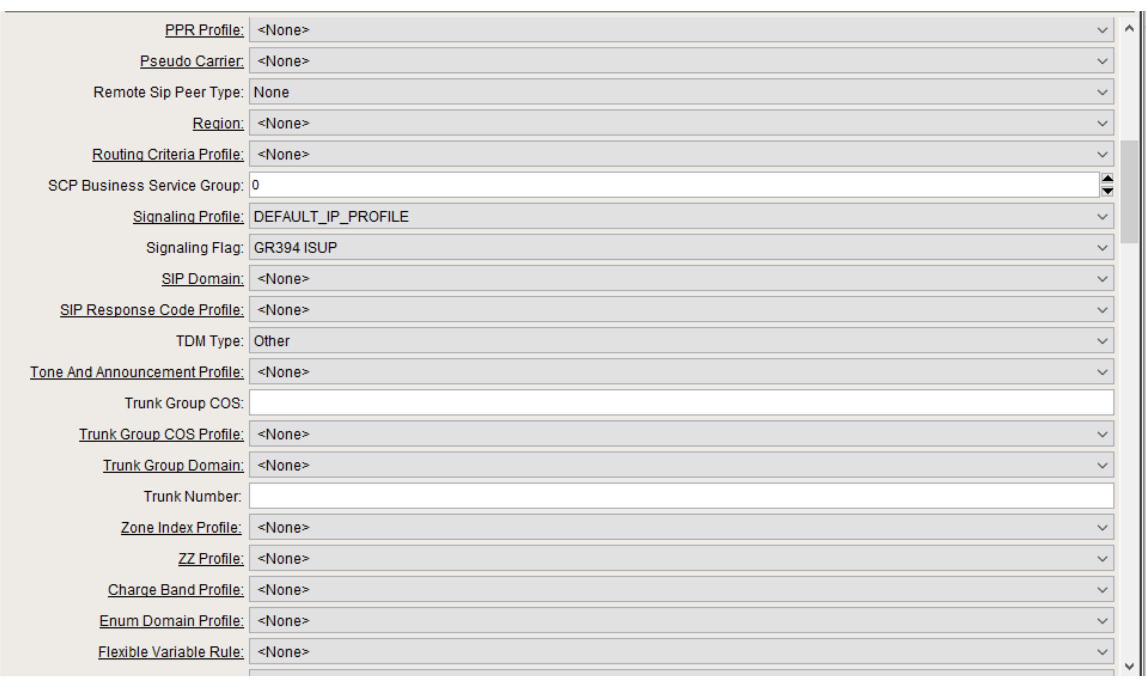


Figure 84:

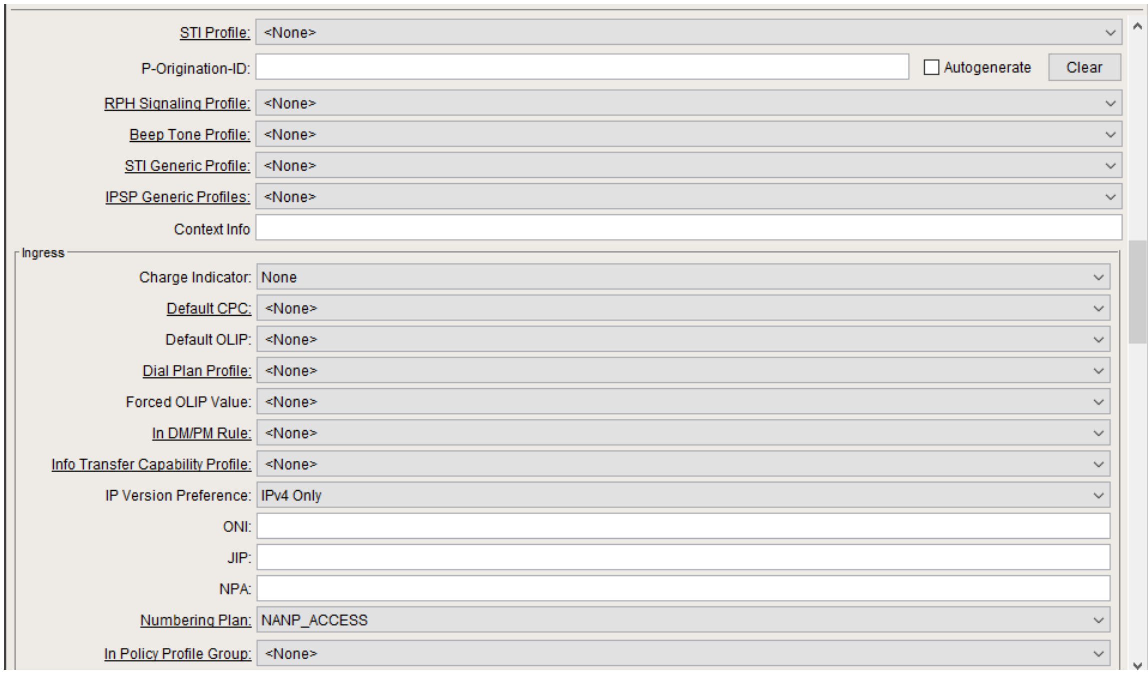


Figure 85:

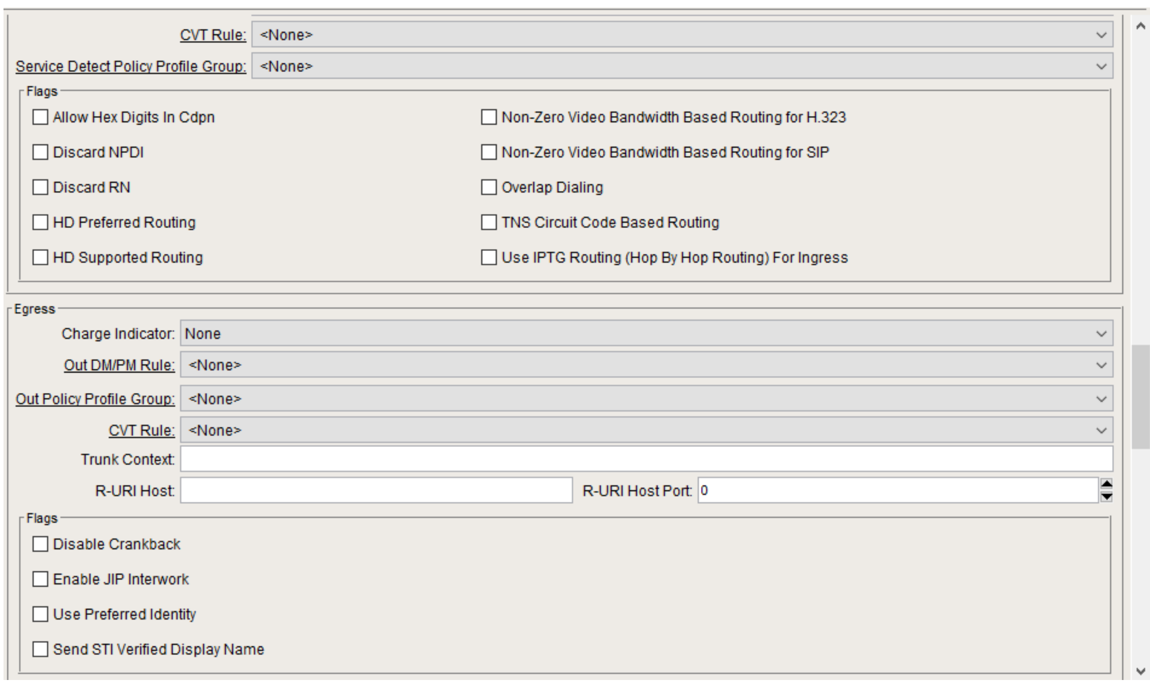


Figure 86:

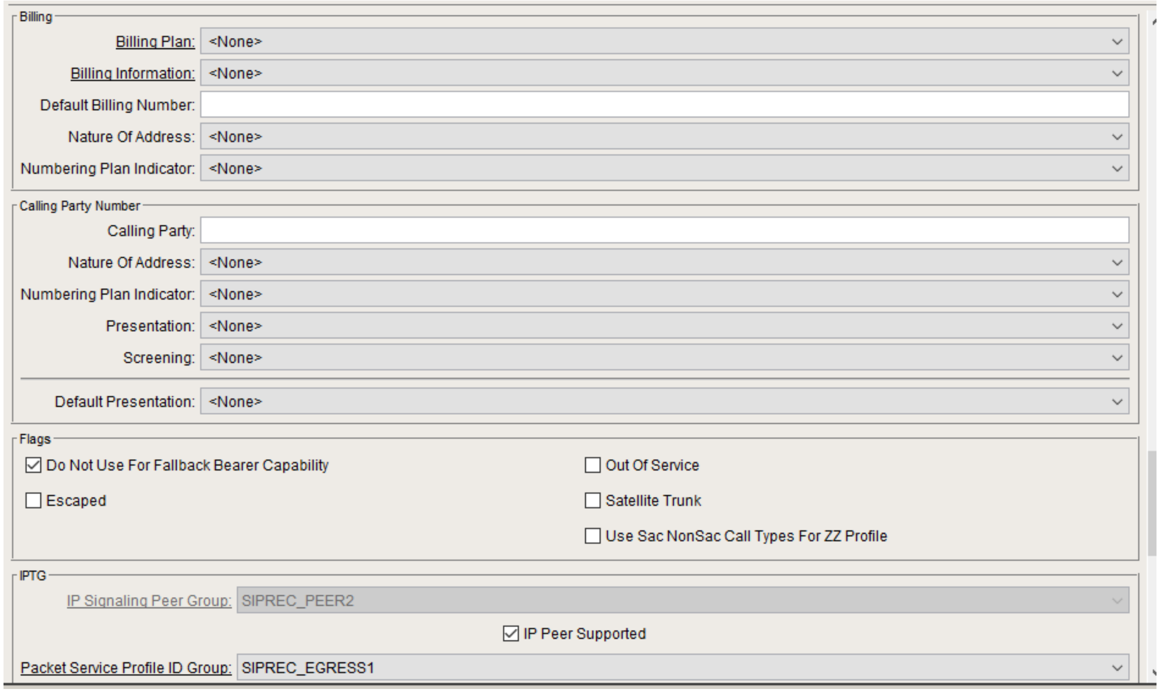
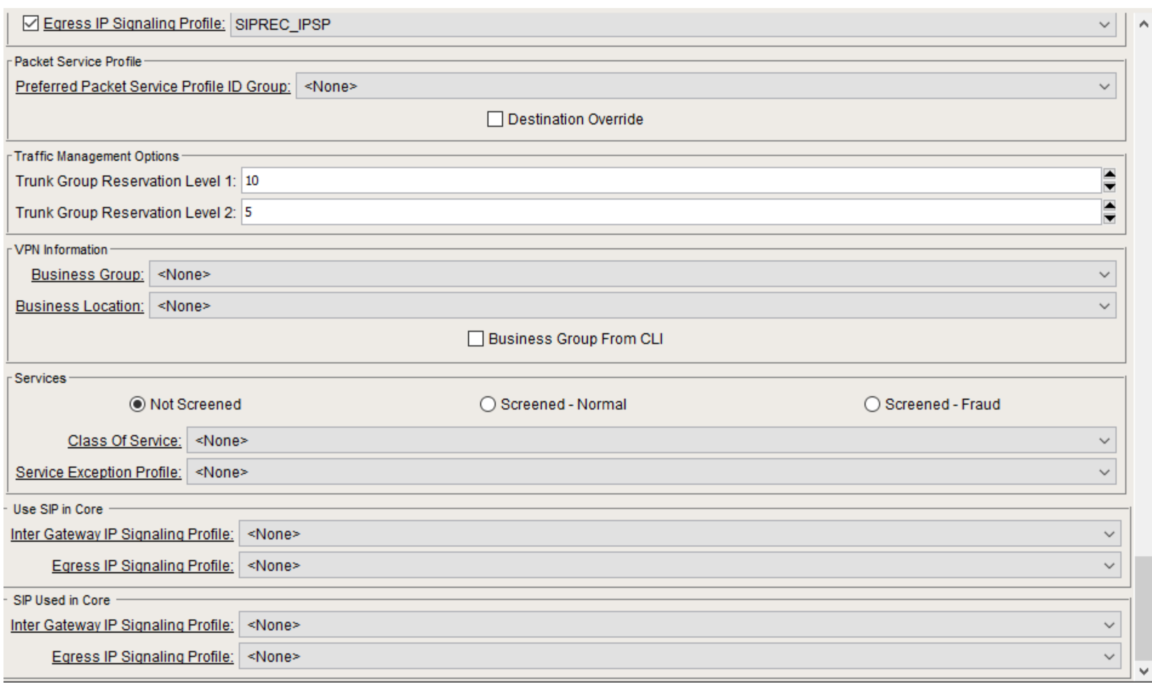


Figure 87:

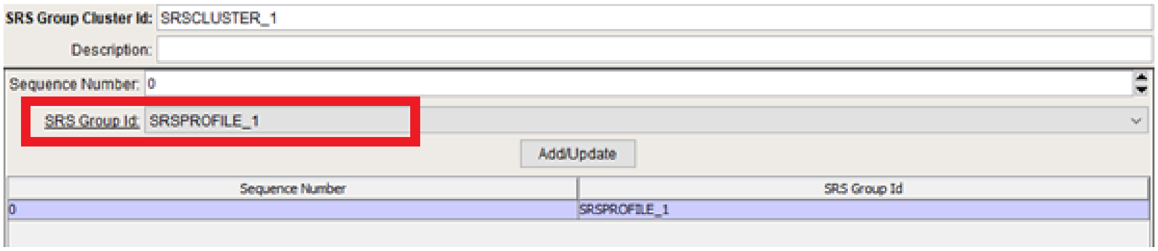


SRS Cluster

An SRS is the target to which the SBC sends session recordings.The SBCsupports configuring multiple SRS' on the PSX using SRS Group Profiles.

SRS Cluster contains multiple SRS Groups for simultaneous recording (up to 4).

Figure 88:



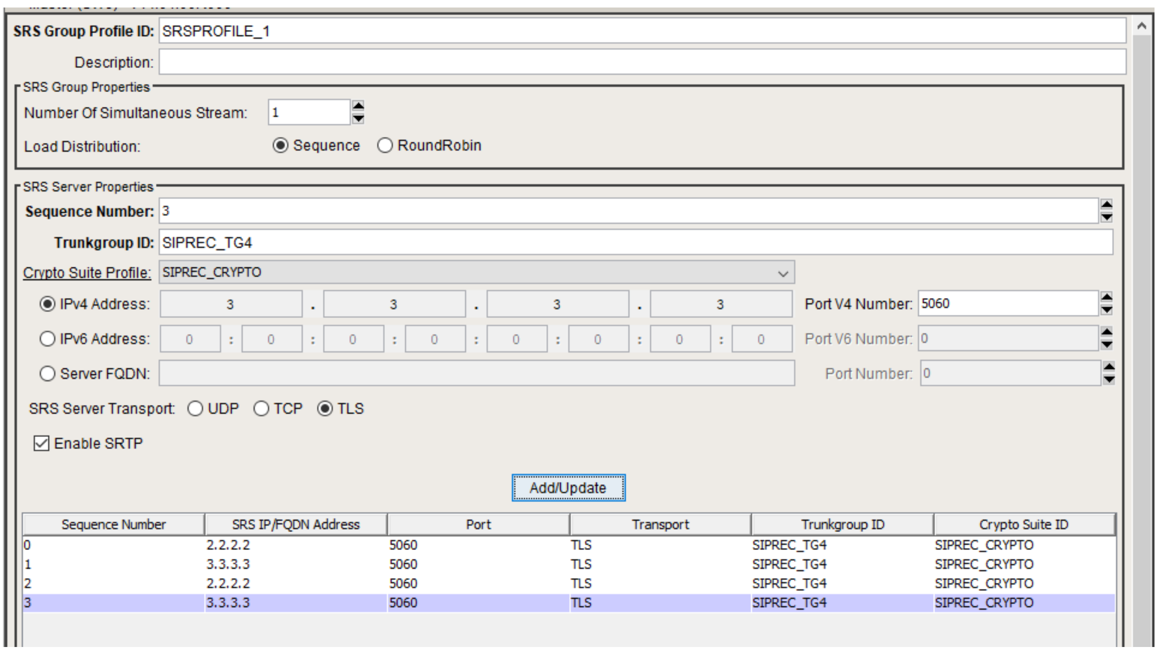
SRS Group Profile

Provide NICE recorder, primary and secondaryIPV4 or IPV6 address and port (5060). Also, mention the NICE TG name. The name of the NICE TG created in the SBC and the PSX shouldbe the same, otherwise recording would not be initiated toward NICE. Transport type can be set to UDP/TCP /TLS. We have to configure appropriate transport at NICE for successful recordings. Please refer toNICE transport configurations for NICE specific configurations

To enable SRTP, we can choose CryptoSuite from the dropdown. For more details referMedia Encryption

1. NICE SRS IP and port details should be configured as per customer deployment.
2. Transport preference mentioned in SRS Group profile should match transport preferences inTrunk Group towards SIPRec zone.

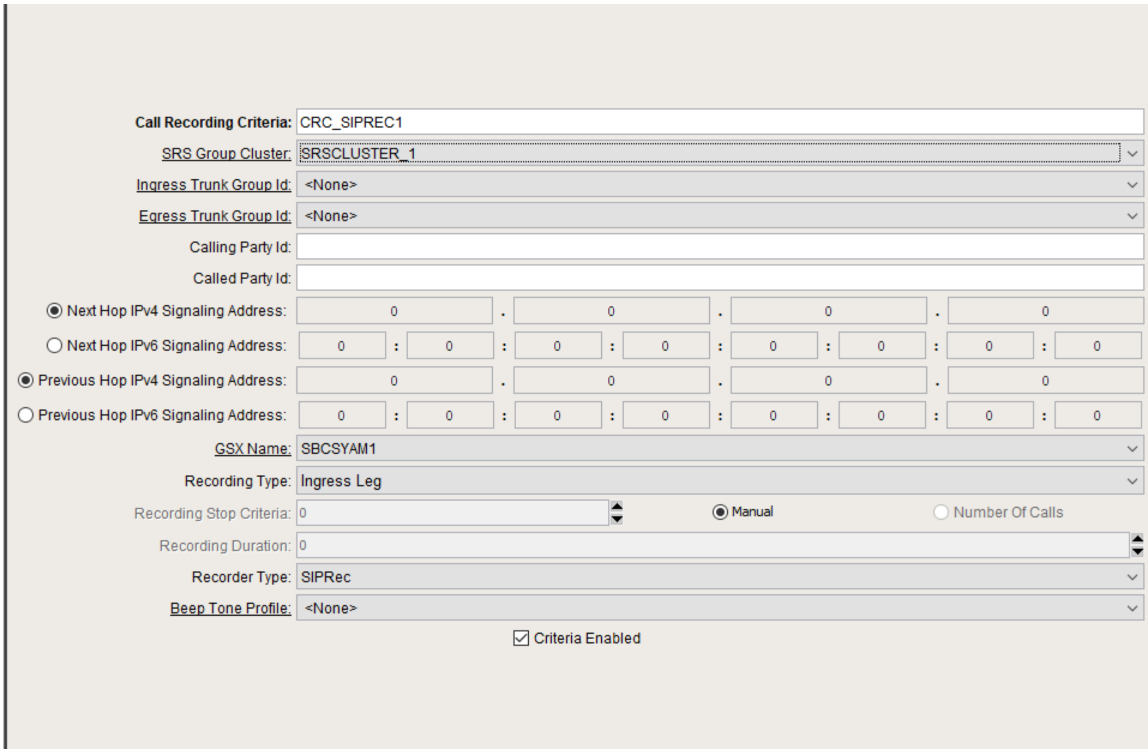
Figure 89:



Call Recording Criteria

Providecall criteria for recording which you wish to record, like calling number,called number, ingress andegress TG, SBC name,the leg you want torecord, and either ingress oregress.Recorder type should be SIPRec. Enable the criteria. When a call is made, it shall be recorded if it falls under this criteria.

Figure 90:

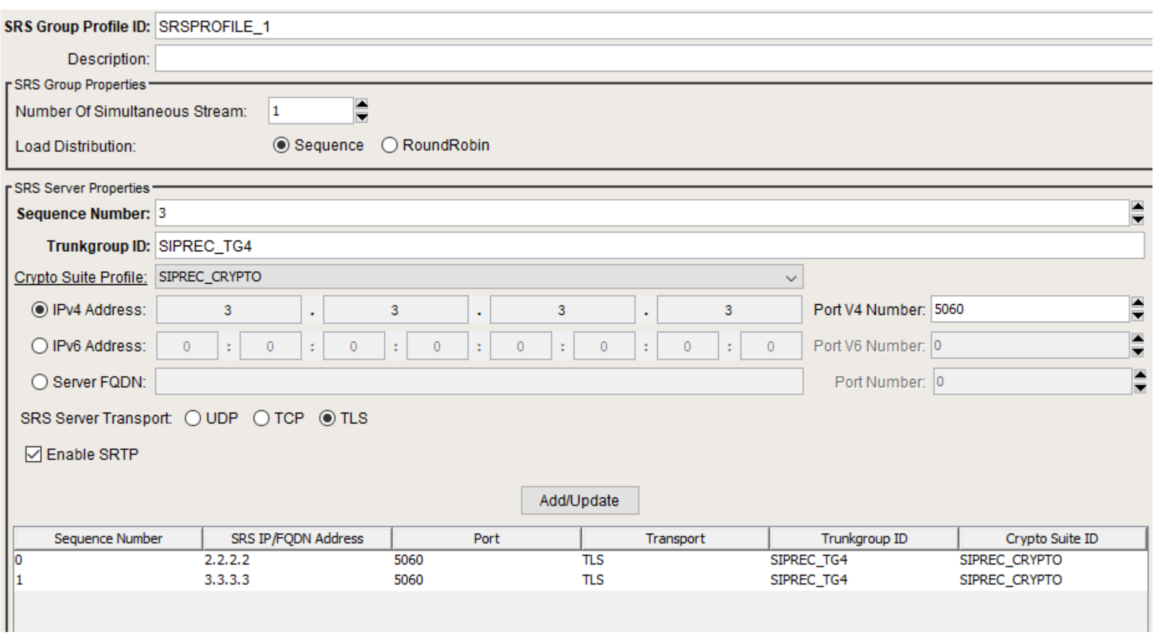


Call Forking to two Active recorders

Configure Number of Simultaneous Stream to "**2**", for SBC to stream media simultaneously to two **Active** SRSs.

> ⓘ Use this configuration only when you have two independent NICE recorder setups with both configured SRSs running in Active mode.

Figure 91:



Redundancy withActive-Standby SRSs

With the SRS redundancy solution, the integration includes two SRS, where one is active (primary-SRS1)and the standby is inactive (secondary-SRS2). If the primary SRS fails, then the secondary SRS becomesactive.

Ribbon recommends NICE to be configured with Failback disabled. Refer toNICE configuration for NoFailback modefor additional NICE configuration changes.

With Failback disabled, If the primary SRS fails, the secondary SRS becomes active. When the primary SRS comes back up, the secondary SRS remains active and the primary server becomes inactive.

Sequential Forking

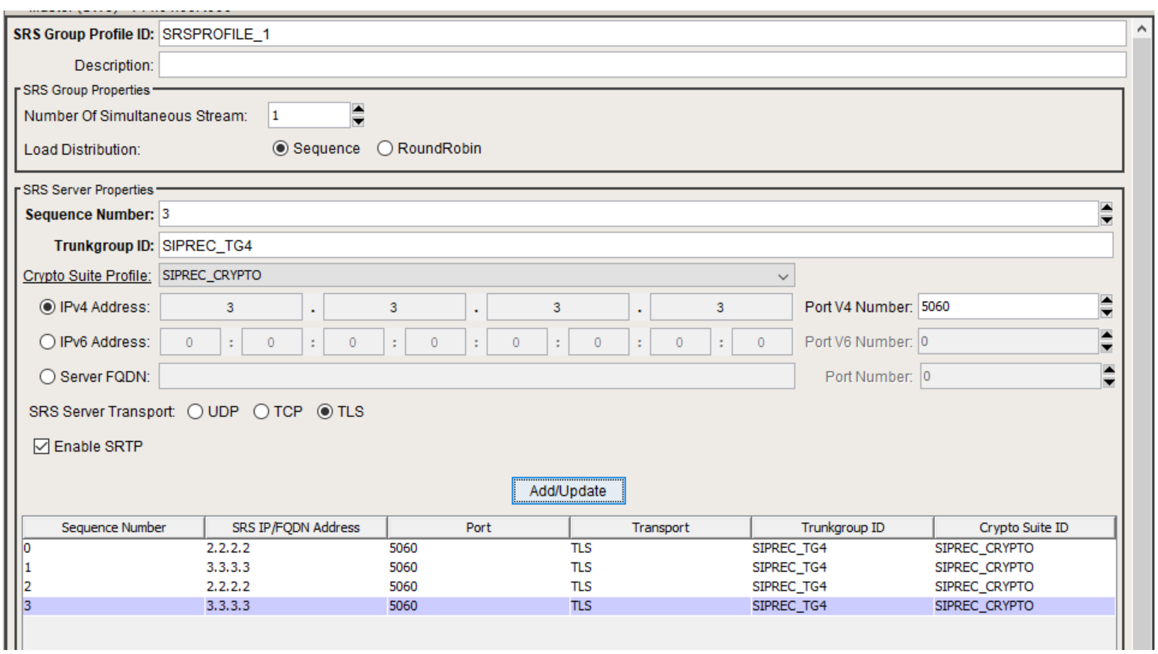
When the number of simultaneous streams is set to 1, the SBC shall start streaming to active SRS with lowest sequence number[SRS1]. If the SRS1 goes down, the SBC blacklists the SRS1 and the SBC automatically uses the next active SRS - SRS2 in the SRS group. Refer toNICE Configurations for Sequential Forkingfor additional NICE configuration changes.

With below pathcheck profile configuration, the SBC blacklists unreachable SRS servers as well as Standby SRS servers[based on 5xx response for OPTIONs]. So, theSBC is responsible for detecting SRS failures and initiating a new session to SRS for both ongoing and new calls.

In the pathcheck profile associated to SRS IP peers, we configure**failureResponseCodes**parameter to define 5xx response codes from Standby SRS server to treat as failure response. So, the SBC blacklists Standby SRS to avoid creating new recording sessions to the inactive SRS.

```
set profiles services pathCheckProfile sip_recording1 protocol sipOptions
set profiles services pathCheckProfile sip_recording1 sendInterval 10
set profiles services pathCheckProfile sip_recording1 replyTimeoutCount 3
set profiles services pathCheckProfile sip_recording1 recoveryCount 1
set profiles services pathCheckProfile sip_recording1 failureResponseCodes [ all5xx ]
set profiles services pathCheckProfile sip_recording1 transportPreference preference1 tls-tcp
comm
```

Figure 92:



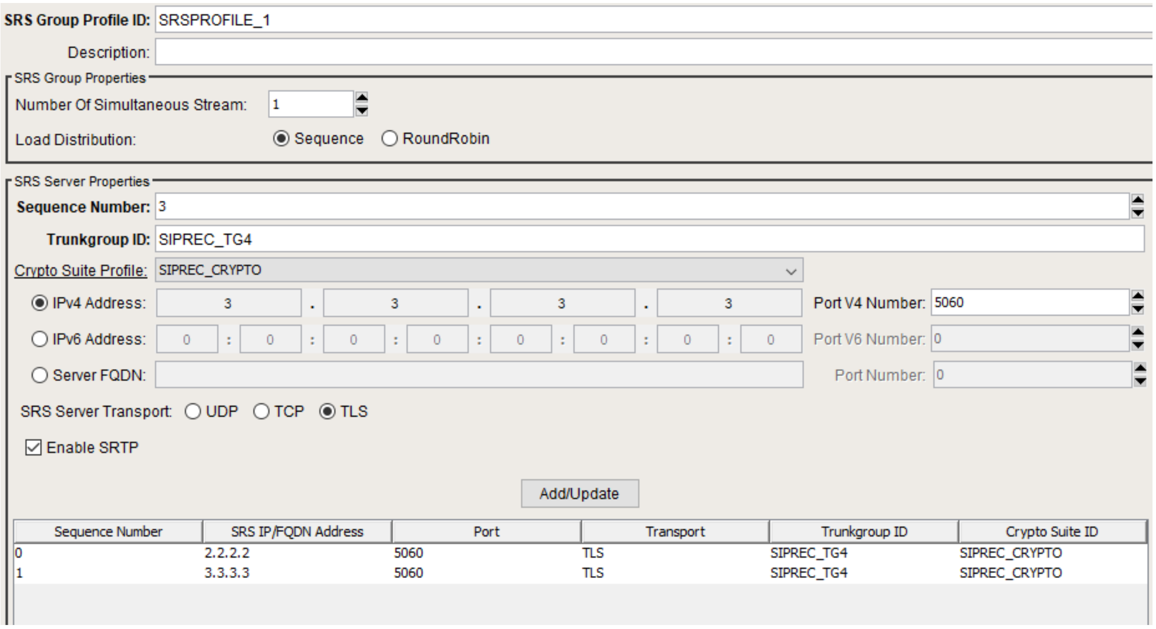
Parallel Forking

When the number of simultaneous stream is set to "2", the SBC sends two streams to primary[SRS1] and secondary[SRS2] SRS.Redundancy is handled by the SRS internally to detect SRS failure and handle the existing sessions. TheSBC connects with SRS1 with active SDP with Active recording and SRS2 with inactive SDP.If SRS1 goes down, SRS2 sends a re-INVITE with active SDP (AIR IP details)to SBC to continue recording via SRS2. ReferNICE Configurations for Parallel Forkingfor additional NICE configuration changes.

In the pathcheck profile associated to SRS IP peers, the SBC blacklists only if there is no response from SRS. Any response from SRS is considered as an active response.

```
set profiles services pathCheckProfile sip_recording1 protocol sipOptions
set profiles services pathCheckProfile sip_recording1 sendInterval 10
set profiles services pathCheckProfile sip_recording1 replyTimeoutCount 3
set profiles services pathCheckProfile sip_recording1 recoveryCount 1
set profiles services pathCheckProfile sip_recording1 transportPreference preference1 tls-tcp
comm
```

Figure 93:



Quad Recording

TheSBC is enhanced to support simultaneously recording SIP egress and ingress legs during a session, for a total of four recordings (four simultaneous streams: two in the ingress leg, and two in the egress leg).

TheSBC provisions the SIP recordings towards all four recorders, two from Ingress tap point and another two from egress tap point. (Due to NP limitations, four simultaneous recordings cannot be triggered on the same call leg.)

- The SBC supports sending the recording streams to up to four SRS servers simultaneously.
- Each recording criteria can be configured with a Recording Cluster. A Recording Cluster can have up to four SRS Groups.
- For Quad SIPREC, there are four recordings triggered. Two recordings are triggered on the Ingress leg and two on the Egress leg.
- If there is more than one SRS Group configured, it is recommended to set`recordingType`to "both legs" or "all legs".
- When SIPREC is selected as the Recorder Type, and Recording Type is selected as both legs and all legs, the SBC by default records the ingress leg.

Create four SRS profiles with one SRS entry in each profile.

Please note we need four NICE recorder setups with all four configured SRSs running in Active mode.

Figure 94:

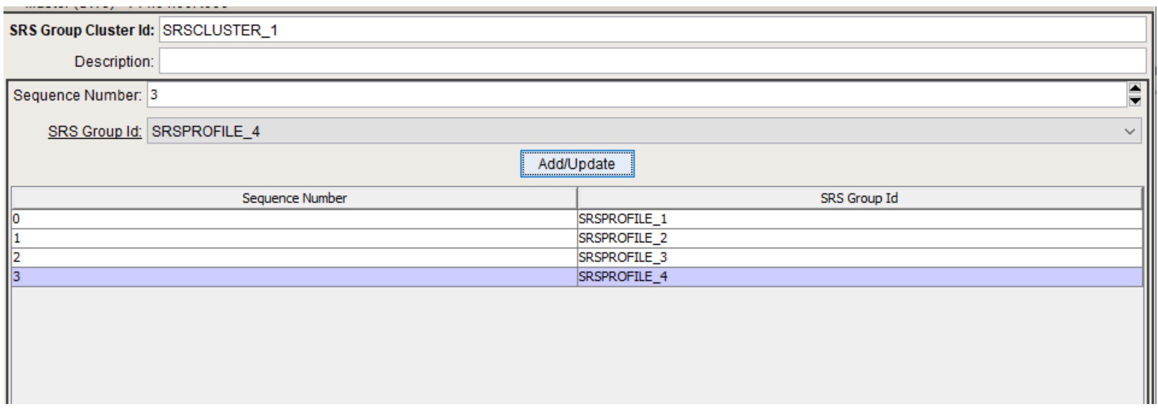
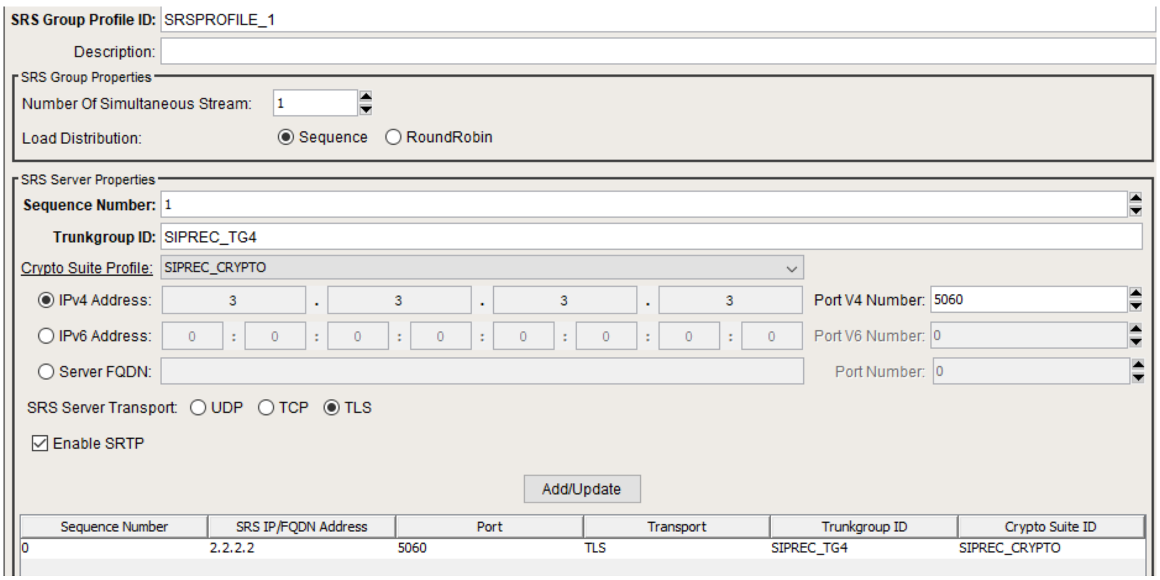


Figure 95:



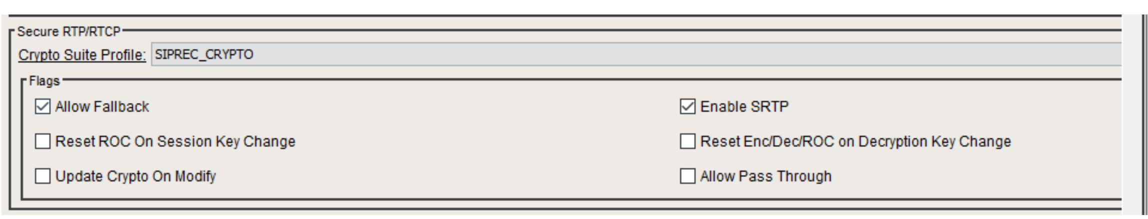
Media Encryption

The Secure Real-time Transport Protocol (Secure RTP or SRTP) is an IETF cryptographic protocol used to provide secure communications over untrusted networks as described in RFC 3711. SRTP provides confidentiality, message authentication, and replay protection to Internet media traffic such as audio and video. TheSBC SWe Core supports Secure RTP and its associated secure real-time transport control protocol (Secure RTCP) for IPv4/IPv6 addressing for both audio and video streams.

Towards Endpoint

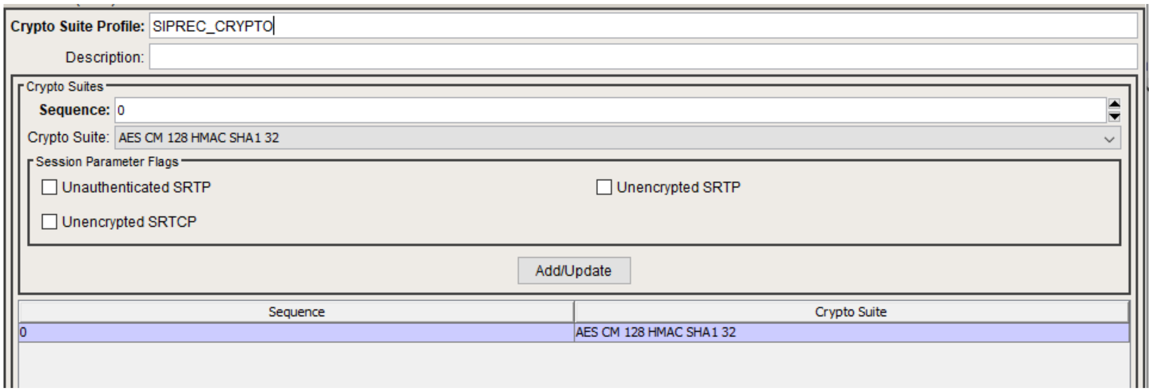
To enable sRTP towards endpoints, Crypto suite profiles must be configured in Packet service profiles mapped towards Ingress and Egress Trunks.

Figure 96:



Add crypto suites to the crypto profile and save it.

Figure 97:

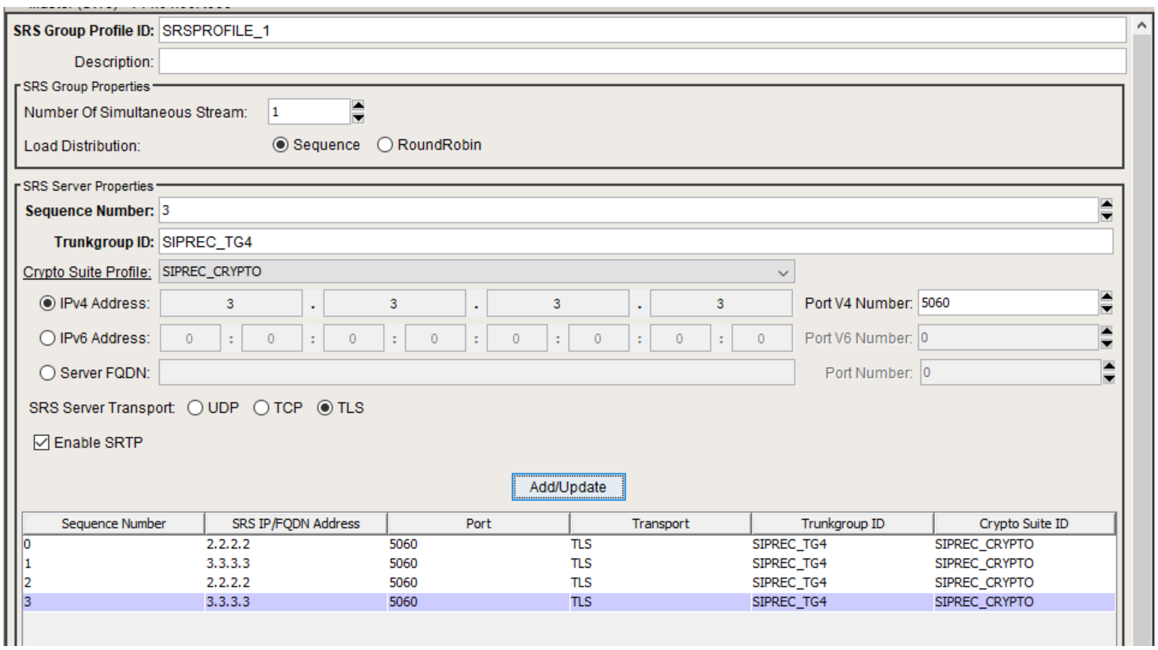


Towards NICE SIP Recorder

To enable encryption towards SIPRec, Crypto suite profiles are attached to SRS Group Profiles.

Check Enable sRTP check box in SRS Profile and select Crypto Suite Profile from the drop down list.

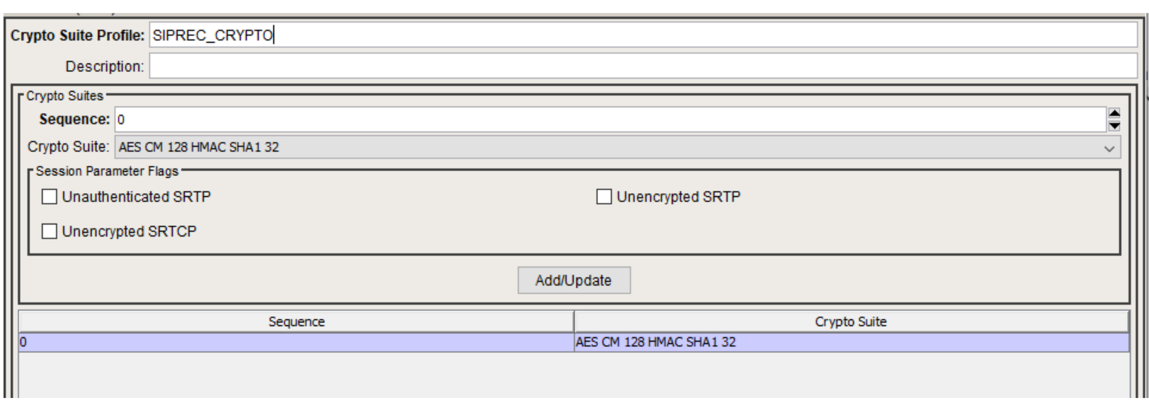
Figure 98:



Add crypto suites to the crypto profile and attach it to the SRS group profile.

NICE supports two sRTP crypto suitesAES_CM_128_HMAC_SHA1_80 andAES_CM_128_HMAC_SHA1_32.

Figure 99:



Ribbon SBC SWe Core High Availability

Info

During this interop, SBC SWe was configured in HA mode with the below configuration for High Availability.

In an HA configuration, the two SBC VMs are connected to each other using the HA ports on the respective VMs. The HA logical ports must be in the same network and routable using the switch and they must be connected to a switch. Failure of the connection is via link detection and also TIPC keep-alives.

HA Configuration Link Detection Group

The Link Detection Group allows you to group interfaces and associated Link Monitors together and track link verification failures within the group. A Link Detection Group (LDG) is configured with a unique name and a failover threshold. The LDG tracks the number of link verification failures that have occurred among the Link Monitors configured.

Create Link Detection Groups for both pkt0 and pkt1 interfaces.

```
set addressContext default linkDetectionGroup pkt0_act_ldg ceName SBCPOOJA1
set addressContext default linkDetectionGroup pkt0_act_ldg type ip
set addressContext default linkDetectionGroup pkt0_act_ldg threshold 1
set addressContext default linkDetectionGroup pkt0_act_ldg state enabled
set addressContext default linkDetectionGroup pkt0_act_ldg linkMonitor pkt0_act_lm interfaceGroup IG1
set addressContext default linkDetectionGroup pkt0_act_ldg linkMonitor pkt0_act_lm interface IF1
set addressContext default linkDetectionGroup pkt0_act_ldg linkMonitor pkt0_act_lm destination
<pkt0_default_gateway>
set addressContext default linkDetectionGroup pkt0_act_ldg linkMonitor pkt0_act_lm state enabled
set addressContext default linkDetectionGroup pkt0_stb_ldg ceName SBCPOOJA2
set addressContext default linkDetectionGroup pkt0_stb_ldg type ip
set addressContext default linkDetectionGroup pkt0_stb_ldg threshold 1
set addressContext default linkDetectionGroup pkt0_stb_ldg state enabled
set addressContext default linkDetectionGroup pkt0_stb_ldg linkMonitor pkt0_stb_lm interfaceGroup IG1
set addressContext default linkDetectionGroup pkt0_stb_ldg linkMonitor pkt0_stb_lm interface IF1
set addressContext default linkDetectionGroup pkt0_stb_ldg linkMonitor pkt0_stb_lm destination
<pkt0_default_gateway>
set addressContext default linkDetectionGroup pkt0_stb_ldg linkMonitor pkt0_stb_lm state enabled
set addressContext default linkDetectionGroup pkt1_act_ldg ceName SBCPOOJA1
set addressContext default linkDetectionGroup pkt1_act_ldg type ip
set addressContext default linkDetectionGroup pkt1_act_ldg threshold 1
set addressContext default linkDetectionGroup pkt1_act_ldg state enabled
set addressContext default linkDetectionGroup pkt1_act_ldg linkMonitor pkt1_act_lm interfaceGroup IG2
set addressContext default linkDetectionGroup pkt1_act_ldg linkMonitor pkt1_act_lm interface IF2
set addressContext default linkDetectionGroup pkt1_act_ldg linkMonitor pkt1_act_lm destination
<pkt1_default_gateway>
set addressContext default linkDetectionGroup pkt1_act_ldg linkMonitor pkt1_act_lm state enabled
set addressContext default linkDetectionGroup pkt1_stb_ldg ceName SBCPOOJA2
set addressContext default linkDetectionGroup pkt1_stb_ldg type ip
set addressContext default linkDetectionGroup pkt1_stb_ldg threshold 1
set addressContext default linkDetectionGroup pkt1_stb_ldg state enabled
set addressContext default linkDetectionGroup pkt1_stb_ldg linkMonitor pkt1_stb_lm interfaceGroup IG2
set addressContext default linkDetectionGroup pkt1_stb_ldg linkMonitor pkt1_stb_lm interface IF2
set addressContext default linkDetectionGroup pkt1_stb_ldg linkMonitor pkt1_stb_lm destination
<pkt1_default_gateway>
set addressContext default linkDetectionGroup pkt1_stb_ldg linkMonitor pkt1_stb_lm state enabled
comm
```

NICE Configuration

For detailed NICE configurations, please visit official NICE support pagehttp://www.extranice.com/.

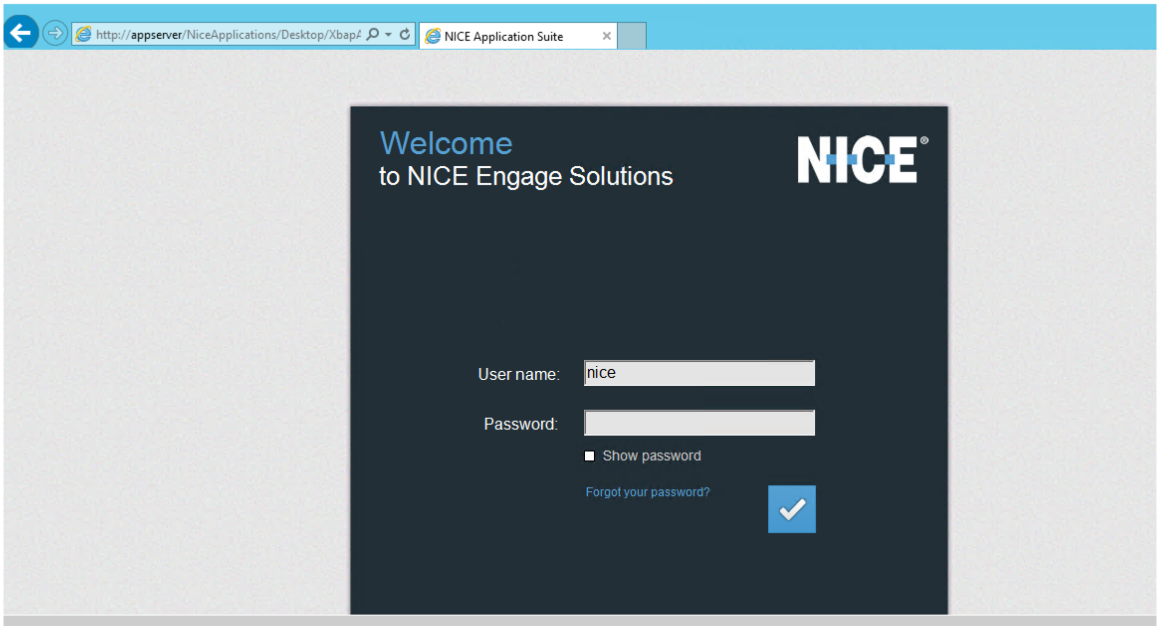
As a part of this document, we have highlighted specific NICE configuration changes that were used in our testing.

Please note that the configurations mentioned below were used in our lab environment for testing purposes. Each customer may have unique needs and configurations. Ribbon recommends that customers work with NICE engineers for NICE configurations to best meets their requirements.

Application Server

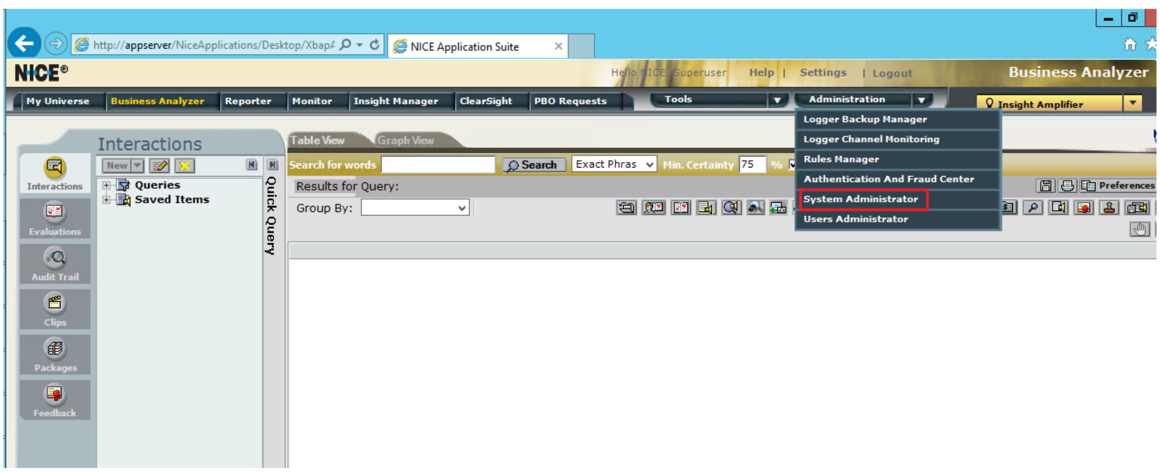
Use the below login page to access NICE application server for all the NICE configurations.

Figure 100:



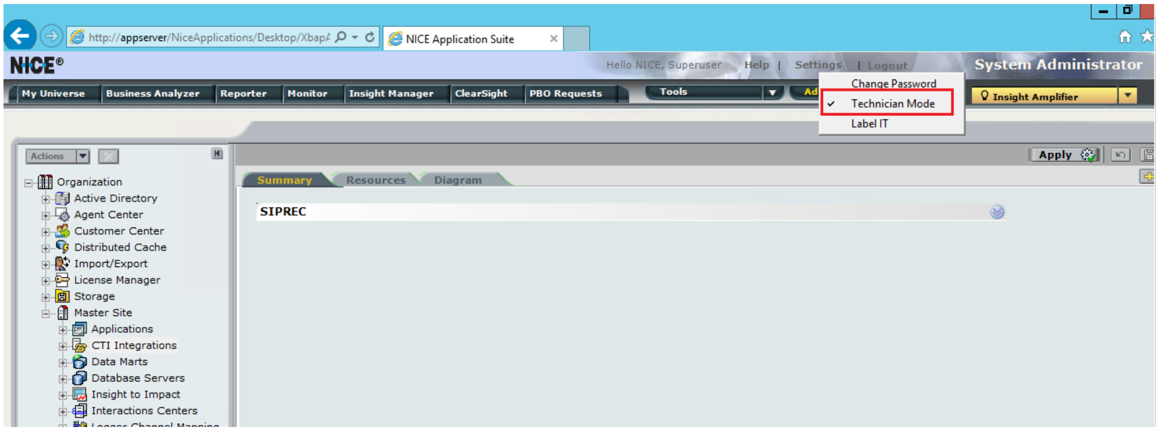
After login, Select System Administrator from the dropdown as mentioned below to check NICE configurations.

Figure 101:



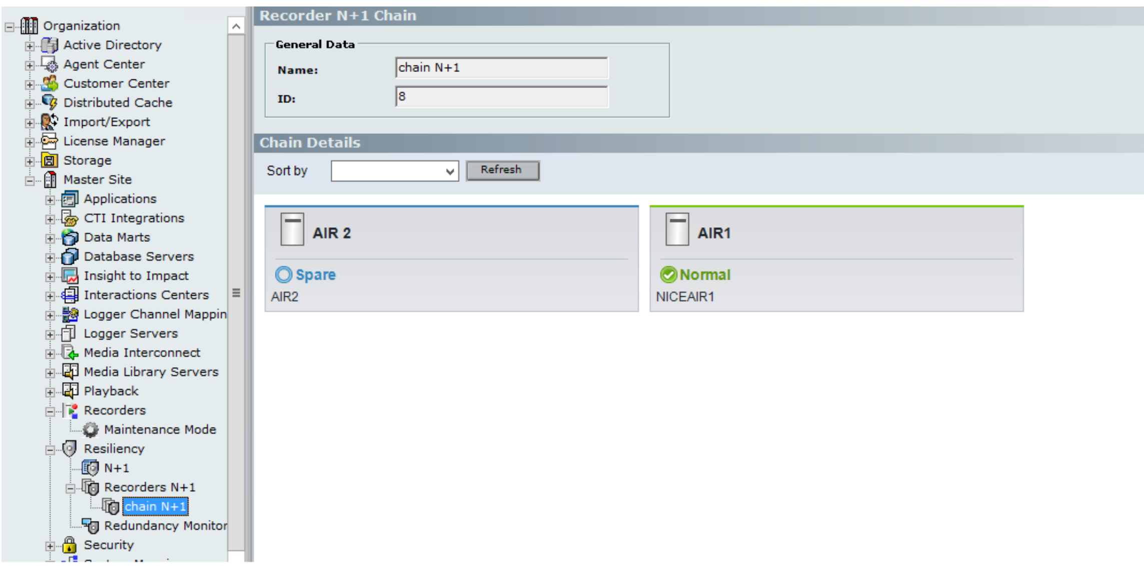
Once logged in as System Administrator, Check Enable Technician Mode from the drop down as mentioned below to edit any configurations.

Figure 102:



To check Active and Standby AIR servers, go to Master Site > Resiliency > Recorders N+1 > chain N+1.

Figure 103:



Metadata Support

Support for Metadata type 'sonus'

When siprecmetadata profile is not configured, by default theSBC supports backward compatibility and pre-defined metadata for passing proprietary call specific information from the SRC to the SRS.

In order to configure NICE server to support default Ribbon SBC configurations, Go to:

- Master Site > CTI Integrations > Media Provider Controllers >Additional Media Provider Controller Parameters >MetadataType > sonus
- Click Save
- In the CTI Integrations branch, click Apply

> - Please repeat the above steps for both the VRSP servers.
> - Restart NICE Integration Dispatch Service on both the VRSP servers.

Support for Metadata type 'RFC7865'

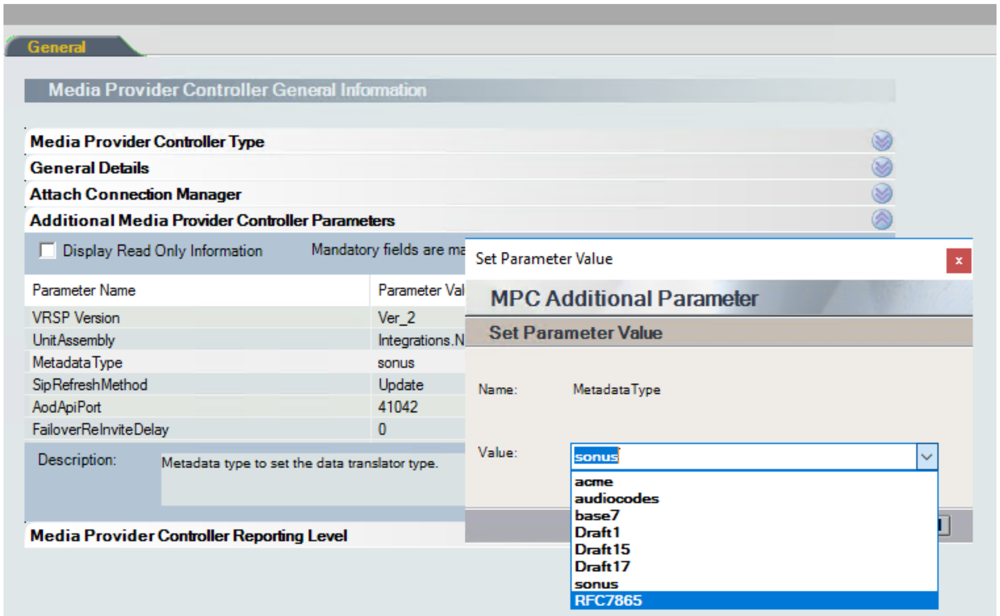
When siprecmetadata profile version is set to 1, Ribbon SBC supports RFC 7865.

In order to configure NICE server to support RFC7865, Go to:

- Master Site > CTI Integrations > Media Provider Controllers >Additional Media Provider Controller Parameters >MetadataType > RFC7865
- Click Save
- In the CTI Integrations branch, click Apply

- Please repeat the above steps for both the VRSP servers.
- Restart NICE Integration Dispatch Service on both the VRSP servers.

Figure 104:



VRSP NoFailback mode

In order to change configuration at NICE server, Go to:

- Master Site >CTI integration > Media Provider Controller tab > VRSP[A/S]CTI integration > Media Provider Controller tab > VRSP[A/S] > RunningMode > NOFAILBACK
- Click Save
- In the CTI Integrations branch, click Apply

- Please repeat the above steps for both the VRSP servers.
- Restart NICE Integration Dispatch Service on both the VRSP servers.

Figure 105:

RedundancyIsEnabled	Yes
SrvPosition	Primary
RunningMode	NOFAILBACK
RedundancyRemoteIpAddress	172.16.106.221
RedundancyRemotePort	50501

Figure 106:

RedundancyIsEnabled	Yes
SrvPosition	Secondary
RunningMode	NOFAILBACK
RedundancyRemoteIpAddress	172.16.106.223
RedundancyRemotePort	50501

Transport Configurations

VRSP configurations

For UDP/TCP,Go to:

- Master Site > CTI Integrations > Media Provider Controllers >Additional Media Provider Controller Parameters >SipStackTlsEnabled > NO
- Click Save
- In the CTI Integrations branch, click Apply

For TLS, Go to:

- Master Site > CTI Integrations > Media Provider Controllers >Additional Media Provider Controller Parameters >SipStackTlsEnabled > YES
- Master Site > CTI Integrations > Media Provider Controllers >Additional Media Provider Controller Parameters > SipStackTlsCertificateSerialNumber > serial number of the NICE VRSP certificate
- Click Save
- In the CTI Integrations branch, click Apply

> - Please repeat the above steps for both the VRSP servers.

Figure 107:

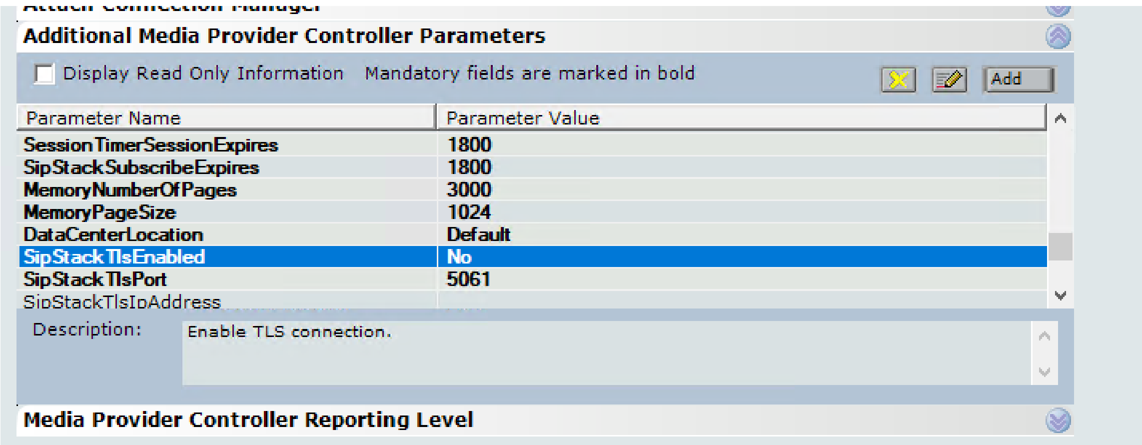
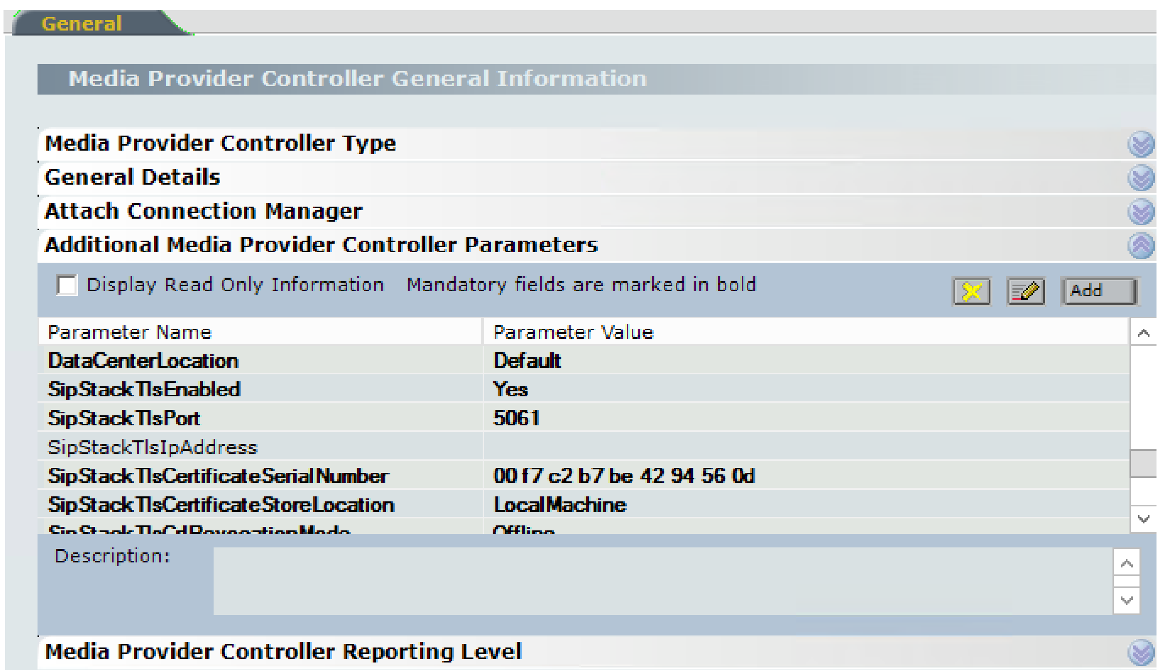


Figure 108:



AIR configurations for UDP with RTP

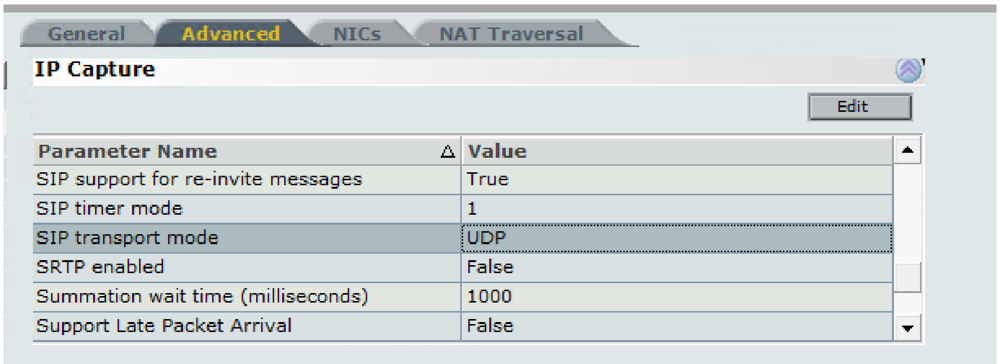
In order to change transport settingsat NICE server, Go to:

- Master Site > Recorders -> AIR[A/S] > Advanced tab > IP Capture >SIP transport mode > UDP
- Master Site > Recorders -> AIR[A/S] >Advanced tab > IP Capture > SRTP enabled > False
- Click Save

- In the CTI Integrations branch, click Apply

> ⓘ
> - Please repeat the above steps for both the AIR servers.

Figure 109:



AIR configurations for TLS with sRTP

In order to change configuration at NICE server, Go to:

- Master Site > Recorders > AIR[A/S] > Advanced tab > IP Capture >SIP transport mode > TLS
- Master Site > Recorders > AIR[A/S] > Advanced tab > IP Capture > SRTP enabled > True
- Master Site > Recorders > AIR[A/S] > Advanced tab > IP Capture > Certificate serial > serial number of the NICE AIR certificate
- Click Save
- In the CTI Integrations branch, click Apply

> ⓘ Please repeat the above steps for both AIR servers.

Figure 110:

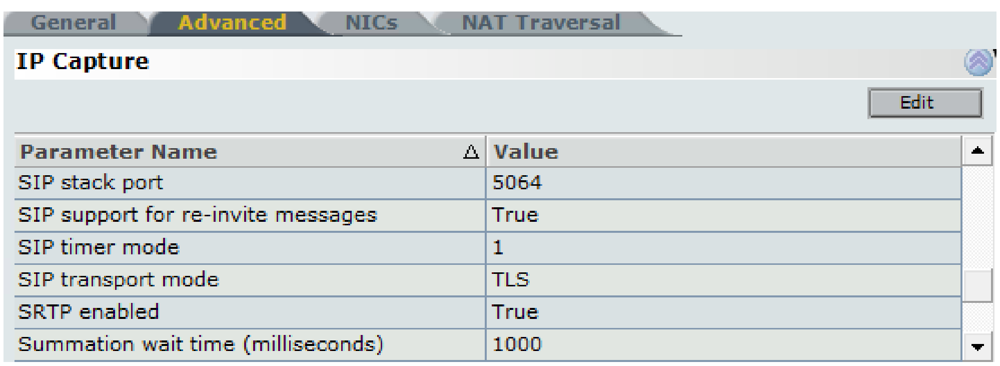
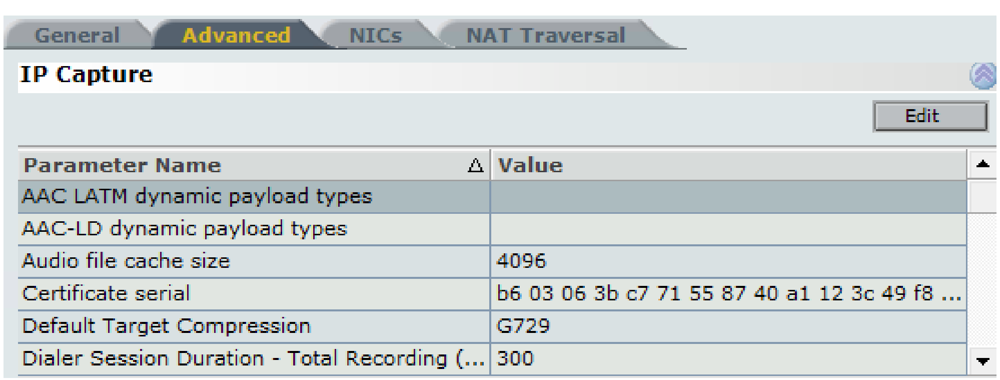


Figure 111:



For Transport changes to be effective:

- Restart NICE Integration Dispatch Service on both the VRSP servers.
- Restart NICE Interactions Center RCM service on Interactions Center server.
- Restart NICE IP Capture and NICE Recorder Administrator services on both AIR servers.

Sequential Forking

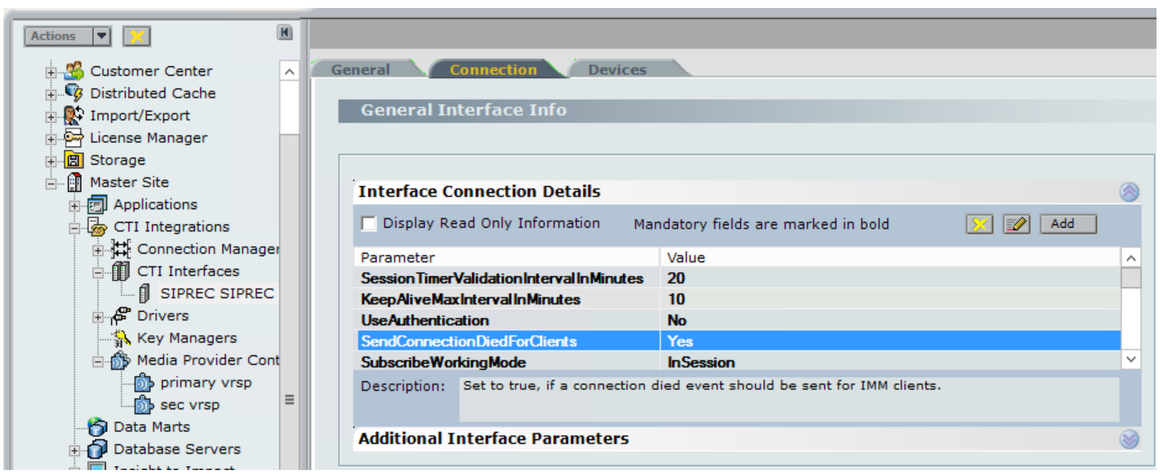
In order to change configuration at NICE server, Go to:

- Master Site > CTI Integrations > Media Provider Controller tab > VRSP[A/S] > Additional Media Provider Controller Parameters > RunningMode > NoFailback
- Master Site > CTI Integrations > CTI Interfaces > Connection > Interface Connection Details > SendConnectionDiedForClients > Yes
- Click Save
- In the CTI Integrations branch, click Apply

For Transport changes to be effective:

- Restart NICE Integration Dispatch Service on both the VRSP servers.
- Restart NICE IP Capture and NICE Recorder Administrator services on both AIR servers.

Figure 112:



Parallel Forking

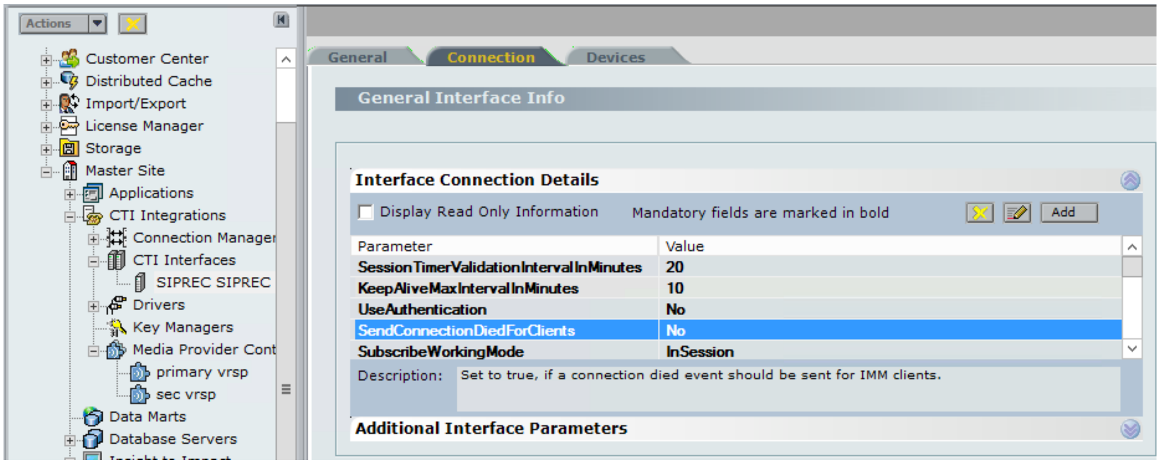
In order to change configurations at NICE server, Go to:

- Master Site > CTI Integrations > Media Provider Controller tab > VRSP[A/S] > Additional Media Provider Controller Parameters > RunningMode > NoFailback
- Master Site > CTI Integrations > CTI Interfaces > Connection > Interface Connection Details > SendConnectionDiedForClients > No
- Master Site > Integration Centers > IC_Server > Configuration > Call Server > op_MaxOpenCallDuration /op_MaxOpenCompoundCallDuration > Desired timeout for long call duration.
- Click Save
- In the CTI Integrations branch, click Apply

For Transport changes to be effective:

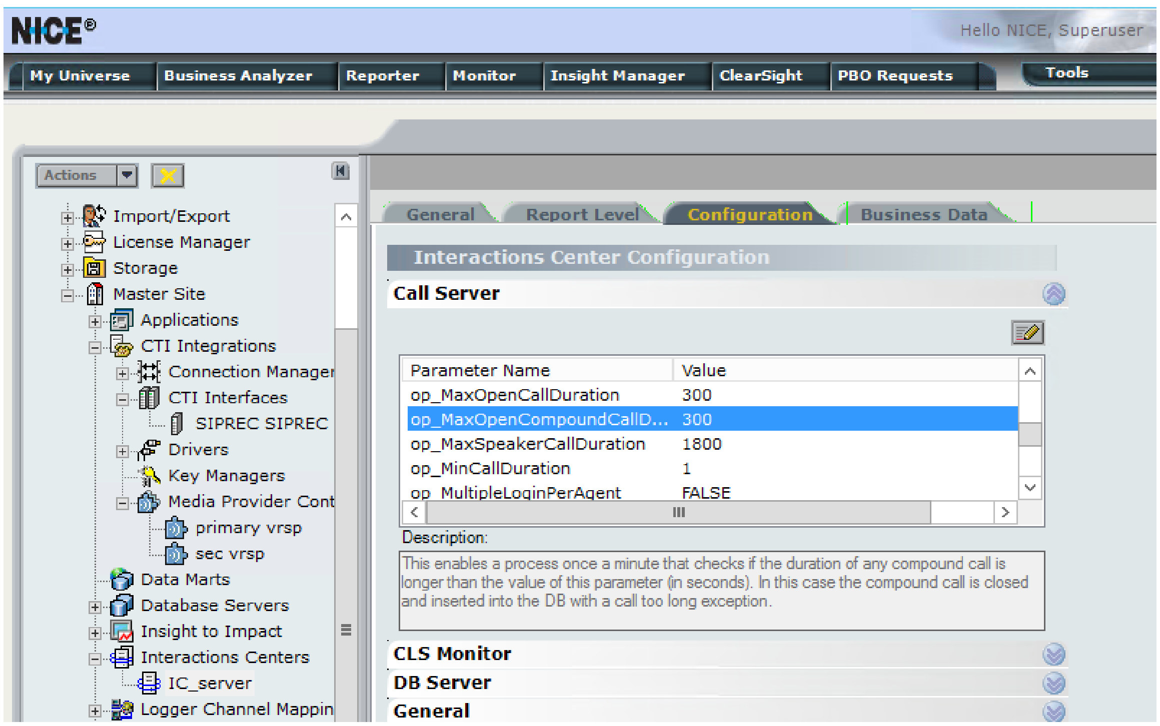
- Restart NICE Integration Dispatch Service on both the VRSP servers.
- Restart NICEInteractions Center Core and NICE Interactions Center RCM services on Interactions Center server.
- Restart NICE IP Capture and NICE Recorder Administrator services on both AIR servers.

Figure 113:



Please note the timeouts captured in the snapshots were configured solely for the purpose of testing. Please tune this timeout as per specific business needs.

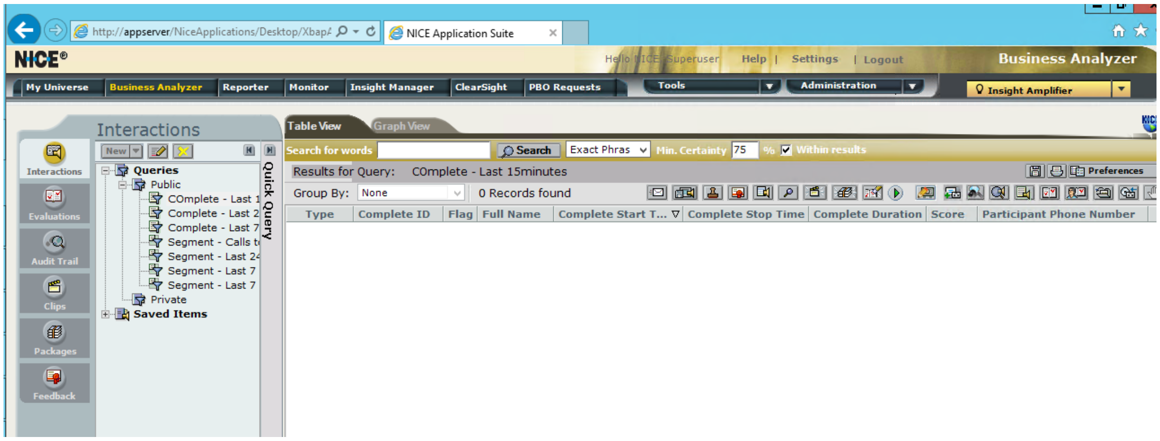
Figure 114:



NICE Business Analyzer

Use NICE Business Analyzer to view/query/listen to recordings created.

Figure 115:



Supplementary Services & Features Coverage

The following checklist depicts the set of services/features covered through the configuration defined in this Interop Guide.

Sr. No	Supplementary Services/ Features	Coverage
1	Basic Call Setup & Termination	✔
2	Call Recording via CLI	✔
3	DTMF - RFC2833/ Inband	✔
4	DTMF - SIP INFO	✘
5	Call Hold/ Resume	✔
6	Call Transfer (Blind/ Unattended)	✔
7	Call Transfer (Attended)	✔
8	Session Refresh	✔
9	Call Forward No Answer	✔
10	Conference	✔
11	Transcoding	✔
12	Music On Hold	✔
13	TLS with SRTP	✔
14	SIPRec Call Forking	✔
15	Quad Recording	✔
16	HA SBC switchover	✔
17	SRS Redundancy - Sequential Forking	✔
18	SRS Redundancy - Parallel Forking	✔

Legend

Supported	✔

Not Supported	✗

Caveats

Ribbon:

- SIPRec leg goes to Inactive state after call transfer with REFER processed on SBC while recording type is set to either "Egress" or "Ingress."
- SBC sends two different session_id's for single call towards Active and Standby NICE SRS servers. During NICE VRSP failover scenarios, NICE recorder is unable to map the two sessions to a single interaction. As a workaround to avoid any recording loss,at NICE, we configure op_MaxOpenCompoundCallDuration/"op_MaxOpenCallDuration". NICE will push open interactions handled by failed SRS server to NBA as a new file after this configured timeout [default 5 hours].

Nice:

- Upon NICE VRSP failover, it may take up to three minutes (default) for AIR to refresh the session and retrieve keys from secondary VRSP. This may result in failure to decrypt and open any new calls for up to three minutes ("white noise" + exception on the interaction).

Support

For any support related queries about this guide, please contact your local Ribbon representative, or use the details below:

- Sales and Support: 1-833-742-2661
- Other Queries: 1-877-412-8867
- Website:https://ribboncommunications.com/services/ribbon-support-portal

References

For detailed information about Ribbon products & solutions, please visit:https://ribboncommunications.com/products

Conclusion

This Interoperability Guide describes successful configuration of Ribbon SBC SWe Core& PSX withNICE SIP Recorder.

All features and capabilities tested are detailed within this document - any limitations, notes or observations are also recorded in order to provide the reader with an accurate understanding of what has been covered, and what has not.

Configuration guidance is provided to enable the reader to replicate the same base setup - there may be additional configuration changes required to suit the exact deployment environment.