Configuration Note

*AudioCodes Professional Services – Interoperability Lab*

# Microsoft® Teams Direct Routing Enterprise Model and Colt SIP Trunk using AudioCodes Mediant™ SBC

Version 7.2

colt
smarter / faster / further

Microsoft Teams

Microsoft Partner
Gold Communications

audiocodes

# Table of Contents

## Notice

This document describes how to connect the Microsoft Teams Direct Routing and Colt SIP Trunk using AudioCodes Mediant SBC product series.

Information contained in this document is believed to be accurate and reliable at the time of printing. However, due to ongoing product improvements and revisions, AudioCodes cannot guarantee accuracy of printed material after the Date Published nor can it accept responsibility for errors or omissions. Updates to this document can be downloaded from https://www.audiocodes.com/library/technical-documents.

This document is subject to change without notice.

**Date Published:** November-07-2018

## WEEE EU Directive

Pursuant to the WEEE EU Directive, electronic and electrical waste must not be disposed of with unsorted waste. Please contact your local recycling authority for disposal of this product.

## Customer Support

Customer technical support and services are provided by AudioCodes or by an authorized AudioCodes Service Partner. For more information on how to buy technical support for AudioCodes products and for contact information, please visit our website at https://www.audiocodes.com/services-support/maintenance-and-support.

## Stay in the Loop with AudioCodes


## Document Revision Record

| LTRT | Description |
|---|---|
| 12315 | Initial document release for Version 7.2. |

## Documentation Feedback

AudioCodes continually strives to produce high quality documentation. If you have any comments (suggestions or errors) regarding this document, please fill out the Documentation Feedback form on our website at https://online.audiocodes.com/documentation-feedback.

**This page is intentionally left blank.**

# 1 Introduction

This Configuration Note describes how to set up the AudioCodes Enterprise Session Border Controller (hereafter, referred to as *SBC*) for interworking between Colt's SIP Trunk and Microsoft's Teams Direct Routing environment.

You can also use AudioCodes' SBC Wizard tool to automatically configure the SBC based on this interoperability setup. However, it is recommended to read through this document to better understand the various configuration options. For more information on AudioCodes' SBC Wizard including the download option, visit AudioCodes Web site at https://www.audiocodes.com/partners/sbc-interoperability-list.

## 1.1 Intended Audience

This document is intended for engineers, or AudioCodes and Colt partners who are responsible for installing and configuring Colt's SIP Trunk and Microsoft's Teams Direct Routing Service in Enterprise Model for enabling VoIP calls using AudioCodes SBC.

## 1.2 About AudioCodes SBC Product Series

AudioCodes' family of SBC devices enables reliable connectivity and security between the Enterprise's and the service provider's VoIP networks.

The SBC provides perimeter defense as a way of protecting Enterprises from malicious VoIP attacks; mediation for allowing the connection of any PBX and/or IP-PBX to any service provider; and Service Assurance for service quality and manageability.

Designed as a cost-effective appliance, the SBC is based on field-proven VoIP and network services with a native host processor, allowing the creation of purpose-built multiservice appliances, providing smooth connectivity to cloud services, with integrated quality of service, SLA monitoring, security and manageability. The native implementation of SBC provides a host of additional capabilities that are not possible with standalone SBC appliances such as VoIP mediation, PSTN access survivability, and third-party value-added services applications. This enables Enterprises to utilize the advantages of converged networks and eliminate the need for standalone appliances.

AudioCodes SBC is available as an integrated solution running on top of its field-proven Mediant Media Gateway and Multi-Service Business Router platforms, or as a software-only solution for deployment with third-party hardware. The SBC can be offered as a Virtualized SBC, supporting the following platforms: Hyper-V, AWS, AZURE, AWP, KVM and VMWare.

## 1.3 About Microsoft Teams Direct Routing

Microsoft Teams Direct Routing allows connecting a customer-provided SBC to the Microsoft Phone System. The customer-provided SBC can be connected to almost any telephony trunk, or connect with third-party PSTN equipment. The connection allows:

- Using virtually any PSTN trunk with Microsoft Phone System
- Configuring interoperability between customer-owned telephony equipment, such as third-party PBXs, analog devices, and Microsoft Phone System

**This page is intentionally left blank.**

# 2 Component Information

## 2.1 AudioCodes SBC Version

**Table 2-1: AudioCodes SBC Version**

| | |
|---|---|
| **SBC Vendor** | AudioCodes |
| **Models** | ▪ Mediant 500 Gateway & E-SBC<br>▪ Mediant 500L Gateway & E-SBC<br>▪ Mediant 800B Gateway & E-SBC<br>▪ Mediant 1000B Gateway & E-SBC<br>▪ Mediant 2600 E-SBC<br>▪ Mediant 4000 SBC<br>▪ Mediant 4000B SBC<br>▪ Mediant 9000 SBC<br>▪ Mediant Software SBC (VE/SE/CE) |
| **Software Version** | 7.20A.204.128 or later |
| **Protocol** | ▪ SIP/UDP (to the Colt SIP Trunk)<br>▪ SIP/TLS (to the Teams Direct Routing) |
| **Additional Notes** | None |

## 2.2 Colt SIP Trunking Version

**Table 2-2: Colt Version**

| | |
|---|---|
| **Vendor/Service Provider** | Colt |
| **SSW Model/Service** | |
| **Software Version** | |
| **Protocol** | SIP |
| **Additional Notes** | None |

## 2.3 Microsoft Teams Direct Routing Version

**Table 2-3: Microsoft Teams Direct Routing Version**

| | |
|---|---|
| **Vendor** | Microsoft |
| **Model** | Teams Phone System Direct Routing |
| **Software Version** | |
| **Protocol** | SIP |
| **Additional Notes** | None |

## 2.4 Interoperability Test Topology

Microsoft Teams Direct Routing can be implemented in the *Enterprise* or *Hosting* Models.

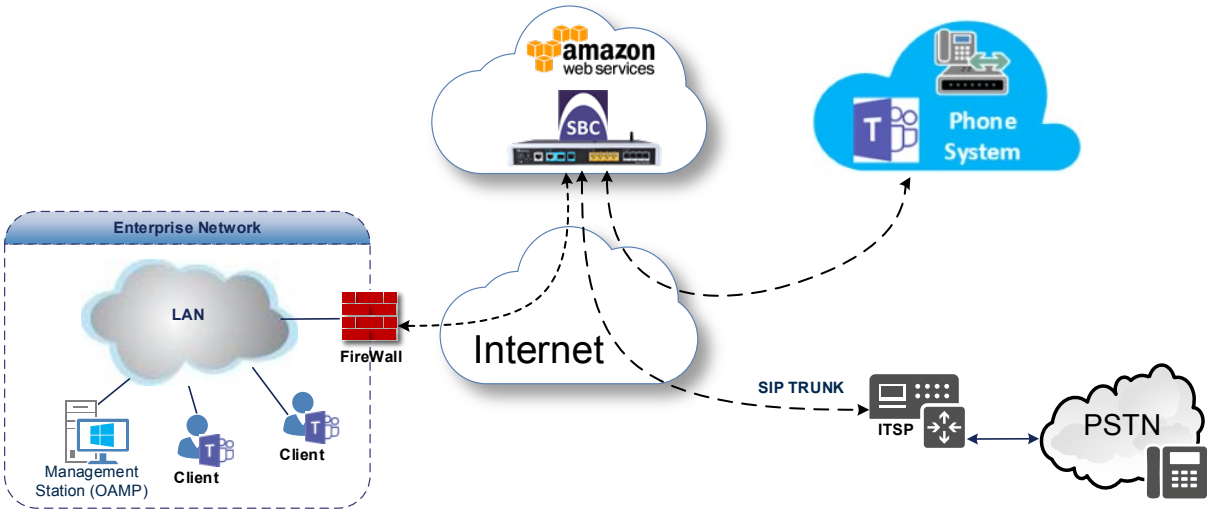
### 2.4.1 Enterprise Model Implementation

The interoperability testing between AudioCodes SBC and Colt SIP Trunk with Teams Direct Routing Enterprise Model was done using the following topology setup:

- Enterprise deployed with Microsoft Teams Phone System Direct Routing Interface located on the WAN for enhanced communication within the Enterprise.
- Enterprise wishes to offer its employees enterprise-voice capabilities and to connect the Enterprise to the PSTN network using Colt's SIP Trunking service.
- AudioCodes SBC, located on the Amazon Web Services Cloud, is implemented to interconnect between the SIP Trunk and Microsoft Teams.
  - **Session:** Real-time voice session using the IP-based Session Initiation Protocol (SIP).
  - **Border:** IP-to-IP network border the Colt's SIP Trunk and Microsoft Teams Phone Systems located in the public network.

The figure below illustrates this interoperability test topology:

**Figure 2-1: Interoperability Test Topology between SBC and Microsoft Teams Direct Routing Enterprise Model with Colt SIP Trunk**



### 2.4.2 Environment Setup

The interoperability test topology includes the following environment setup:

**Table 2-4: Environment Setup**

| Area | Setup |
|---|---|
| **Network** | ▪ Microsoft Teams Direct Routing environment as well as Colt SIP Trunk are located on the Enterprise's (or Service Provider's) WAN |
| **Signaling Transcoding** | ▪ Microsoft Teams Direct Routing operates with SIP-over-TLS transport type<br>▪ Colt SIP Trunk operates with SIP-over-UDP transport type |
| **Codecs Transcoding** | ▪ Microsoft Teams Direct Routing supports G.711A-law, G.711U-law, G.729, G.722, SILK (NB and WB) and OPUS coders<br>▪ Colt SIP Trunk supports G.711A-law, G.711U-law, and G.729 coders |
| **Media Transcoding** | ▪ Microsoft Teams Direct Routing operates with SRTP media type<br>▪ Colt SIP Trunk operates with RTP media type |

### 2.4.3 Infrastructure Prerequisites

The table below shows the list of infrastructure prerequisites for deploying Microsoft Teams Direct Routing.

**Table 2-5: Infrastructure Prerequisites**

| Infrastructure Prerequisite | Details |
|---|---|
| Certified Session Border Controller (SBC) | See Microsoft's document *Deploying Direct Routing Guide.* |
| SIP Trunks connected to the SBC | |
| Office 365 Tenant | |
| Domains | |
| Public IP address for the SBC | |
| Fully Qualified Domain Name (FQDN) for the SBC | |
| Public DNS entry for the SBC | |
| Public trusted certificate for the SBC | |
| Firewall ports for Direct Routing Signaling | |
| Firewall IP addresses and ports for Direct Routing Media | |
| Media Transport Profile | |
| Firewall ports for Teams Clients Media | |

### 2.4.4 Known Limitations

Following limitation was observed in the interoperability tests done for the AudioCodes SBC interworking between Microsoft Teams Direct Routing and Colt 's SIP Trunk:

- During Blind Transfer, the Referred-By header (RFC3892) is used to capture the Calling Line Identifier (CLI) of the Teams party that is transferring the call. Since a CLI in the Referred-By header cannot be screened by Colt, the party to which the call is transferred, will see the default CLI configured on the Colt SIP Trunk.

# 3 Configuring Teams Direct Routing

This section describes how to configure Microsoft Teams Direct Routing to operate with AudioCodes SBC.

## 3.1 Prerequisites

Before you begin configuration, make sure you have the following for every SBC you want to pair:

- Public IP address
- FQDN name matching SIP addresses of the users
- Public certificate, issued by one of the supported CAs

## 3.2 SBC Domain Name in the Teams Enterprise Model

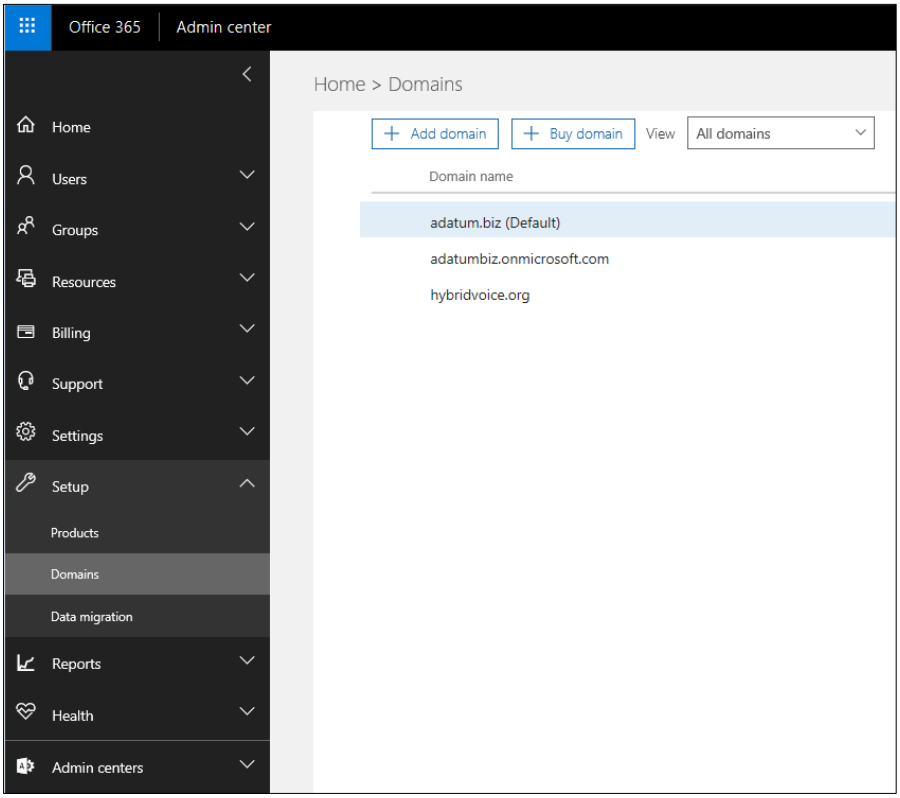
The SBC domain name must be from one of the names registered in 'Domains' of the tenant. You cannot use the ***.onmicrosoft.com** tenant for the domain name. For example, in Figure 2-2, the administrator registered the following DNS names for the tenant:

**Table 3-1: DNS Names Registered by an Administrator for a Tenant**

| DNS name | Can be used for SBC FQDN | Examples of FQDN names |
|---|---|---|
| ACeducation.info | Yes | **Valid names**:<br>▪ sbc.ACeducation.info<br>▪ ussbcs15.ACeducation.info<br>▪ europe.ACeducation.info<br>**Invalid name:**<br>sbc1.europe.ACeducation.info (requires registering domain name europe.atatum.biz in 'Domains' first) |
| adatumbiz.onmicrosoft.com | No | Using ***.onmicrosoft.com** domains is not supported for SBC names |
| hybridvoice.org | Yes | **Valid names**:<br>▪ sbc1.hybridvoice.org<br>▪ ussbcs15.hybridvoice.org<br>▪ europe.hybridvoice.org<br>**Invalid name:**<br>sbc1.europe.hybridvoice.org (requires registering domain name europe. hybridvoice.org in 'Domains' first |

Users can be from any SIP domain registered for the tenant. For example, you can provide users user@ACeducation.info with the SBC FQDN **sbc1.hybridvoice.org** so long as both names are registered for this tenant.

**Figure 3-1: Example of Registered DNS Names**




# 3.3 Example of the Office 365 Tenant Direct Routing Configuration

## 3.3.1 Online PSTN Gateway Configuration

Use following PowerShell command for creating new Online PSTN Gateway:

***New-CsOnlinePSTNGateway*** *-Fqdn* ***sbc.aceducation.info*** *-SipSignallingPort* ***5068*** *-ForwardCallHistory $True MediaBypass $True -Enabled $True*

## 3.3.2 Online PSTN Usage Configuration

Use following PowerShell command for creating an empty PSTN Usage:

***Set-CsOnlinePstnUsage*** *-Identity Global -Usage @{Add=****"Interop"****}*

## 3.3.3 Online Voice Route Configuration

Use following PowerShell command for creating new Online Voice Route and associate it with PSTN Usage:

***New-CsOnlineVoiceRoute*** *-Identity* ***"audc-interop"*** *-NumberPattern "^\+" -OnlinePstnGatewayList* ***sbc.aceducation.info*** *-Priority 1 -OnlinePstnUsages* ***"Interop"***

## 3.3.4 Online Voice Routing Policy Configuration

Use following PowerShell command for assigning the Voice Route to the PSTN Usage:

***New-CsOnlineVoiceRoutingPolicy "audc-interop"*** *-OnlinePstnUsages* ***"Interop"***

### 3.3.5 Enable Online User

Use following PowerShell command for enabling online user:

***Set-CsUser*** *-Identity* ***user1@company.com*** *-EnterpriseVoiceEnabled $true -HostedVoiceMail $true -OnPremLineURI tel:****+12345678901***

### 3.3.6 Assigning Online User to the Voice Route

Use following PowerShell command for assigning online user to the Voice Route:

***Grant-CsOnlineVoiceRoutingPolicy*** *-PolicyName* ***"audc-interop"*** *-Identity* ***user1@company.com***

Use the following command on the Microsoft Teams Direct Routing Management Shell after reconfiguration to verify correct values:

- **Get-CsOnlinePSTNGateway**

```
Identity                : sbc.ACeducation.info
Fqdn                    : sbc.ACeducation.info
SipSignallingPort       : 5068
CodecPriority           : SILKWB, SILKNB, PCMU, PCMA
ExcludedCodecs          :
FailoverTimeSeconds     : 10
ForwardCallHistory      : True
ForwardPai              : False
SendSipOptions          : True
MaxConcurrentSessions   :
Enabled                 : True
MediaBypass             : True
```

**This page is intentionally left blank.**

# 4 Configuring AudioCodes SBC

This section provides step-by-step procedures on how to configure AudioCodes SBC for interworking between Microsoft Teams Direct Routing and the Colt SIP Trunk. These configuration procedures are based on the interoperability test topology described in Section 2.4 on page 10.

This configuration is done using the SBC's embedded Web server (hereafter, referred to as *Web interface*).

**Notes:**

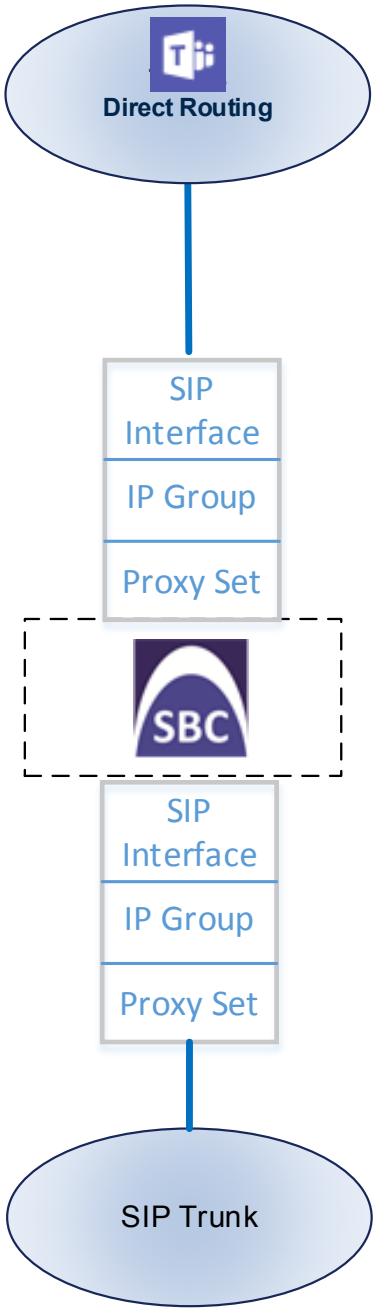
- The scope of this interoperability test and document does **not** cover aspects for configuring Amazon infrastructure for AudioCodes SBC CE. For installation recommendations on AudioCodes' products, refer to the *Mediant Cloud Edition SBC Installation Manual* document, which can be found at AudioCodes web site.
- For implementing Microsoft Teams Direct Routing and Colt SIP Trunk based on the configuration described in this section, AudioCodes SBC must be installed with a License Key that includes the following software features:
  - √ **Microsoft Teams**
  - √ **Security**
  - √ **DSP**
  - √ **RTP**
  - √ **SIP**
  - √ **Number of SBC sessions** *[Based on requirements]*
  - √ **Transcoding sessions** *[If media transcoding is needed]*
  - √ **SILK and OPUS coders** *[Based on requirements]*

  For more information about the License Key, contact your AudioCodes sales representative.
- The scope of this interoperability test and document does **not** cover all security aspects for configuring this topology. Comprehensive security measures should be implemented per your organization's security policies. Especially when AudioCodes SBC implemented as Cloud Edition. For security recommendations on AudioCodes' products, refer to the *Recommended Security Guidelines* document, which can be found at AudioCodes web site

## 4.1 SBC Configuration Concept in Teams Direct Routing Enterprise Model

The diagram below represents AudioCodes' device configuration concept in the Enterprise Model.

**Figure 4-1: SBC Configuration Concept**



## 4.2 IP Network Interface Configuration

This step describes how to configure the SBC's IP network interfaces. There are several ways to deploy the SBC; however, **this** interoperability test topology employs the following deployment method:

- SBC implemented in the Amazon with one IP interface, used for all purposes:
  - Management (OAMP)
  - Signaling and media connectivity to Colt SIP Trunk and Teams Direct Routing

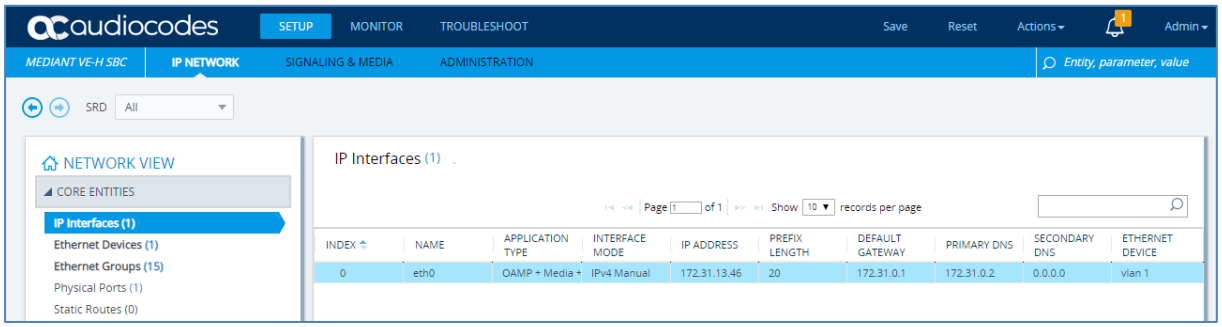
### 4.2.1 Configure VLANs

Default VLANs configuration was used in this interoperability test topology. So, no additional configuration is needed.

### 4.2.2 Configure Network Interface

Network Interface configured automatically in the Amazon implementation. Refer to the *Mediant Cloud Edition SBC Installation Manual* document, which can be found at AudioCodes web site, to configure Amazon image (AMI). The example of the configured IP network interface are shown below:

**Figure 4-2: Configured Network Interface in IP Interfaces Table**




### 4.2.3 Configure NAT Translation

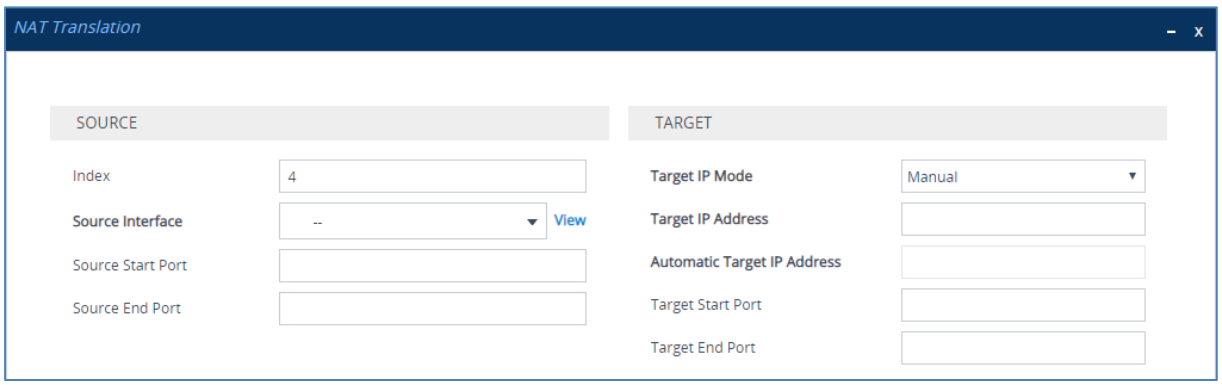
The SBC, located in the Amazon Cloud, implement private IP addresses. The NAT Translation table lets you configure network address translation (NAT) rules for translating source IP addresses into NAT IP addresses (*global - public*), used in front of the Amazon firewall facing the Colt SIP Trunk and Teams Direct Routing.

➢ **To configure the NAT translation rules:**

1. Open the NAT Translation table (**Setup** menu > **IP Network** tab > **Core Entities** folder > **NAT Translation**).

2. Click **New**; the following dialog appears:

**Figure 4-3: NAT Translation Table - Dialog Box**

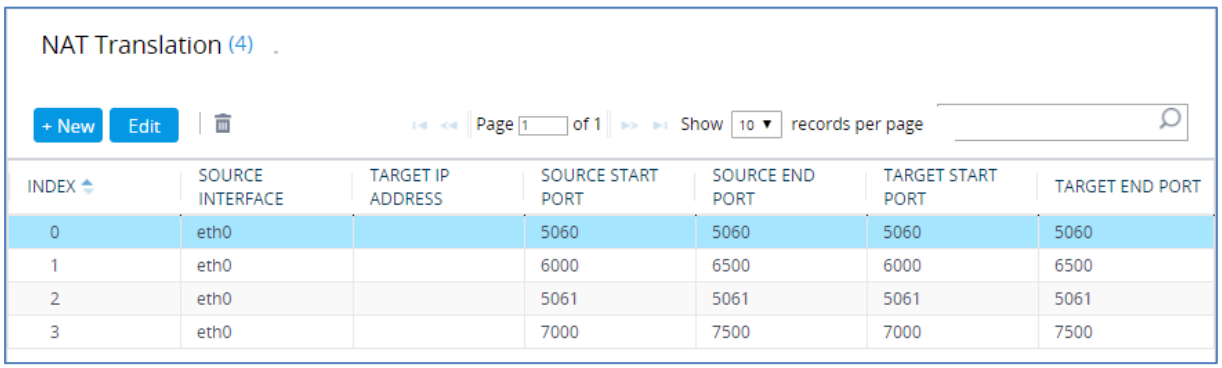



3. Use the following table as reference when configuring a NAT translation rule:

| Parameter | Value |
|---|---|
| Index | **0** |
| Source Interface | **eth0** (IP Network Interface, configured in the previous section) |
| Source Start Port | **5060** (signaling port towards SIP Trunk) |
| Source End Port | **5060** (signaling port towards SIP Trunk) |
| Target IP Mode | **Automatic** (this mode is required if your AWS environment has been configured with an Elastic IP address and you want the device to automatically associate it with the selected source interface as the global (public) IP address). |
| Target IP Address | Configured only if the previous parameter is configured with 'Manual' value. |
| Automatic Target IP Address | Read-only-field |
| Target Start Port | **5060** (signaling port towards SIP Trunk) |
| Target End Port | **5060** (signaling port towards SIP Trunk) |

4. Click Apply.
5. Configure additional rules for Signaling traffic towards Microsoft Teams and Media traffic towards both, SIP Trunk and Microsoft Teams.

**Figure 4-4: Example of the NAT Translation Table**




| INDEX | SOURCE INTERFACE | TARGET IP ADDRESS | SOURCE START PORT | SOURCE END PORT | TARGET START PORT | TARGET END PORT |
|---|---|---|---|---|---|---|
| 0 | eth0 | | 5060 | 5060 | 5060 | 5060 |
| 1 | eth0 | | 6000 | 6500 | 6000 | 6500 |
| 2 | eth0 | | 5061 | 5061 | 5061 | 5061 |
| 3 | eth0 | | 7000 | 7500 | 7000 | 7500 |

# 4.3 SIP TLS Connection Configuration

This section describes how to configure the SBC for using a TLS connection with the Microsoft Teams Direct Routing Phone System. This configuration is essential for a secure SIP TLS connection. The configuration instructions in this section are based on the following domain structure that must be implemented as part of the certificate which must be loaded to the host SBC:

- CN: ACeducation.info
- SAN: *.customers.ACeducation.info

This certificate module is based on the Service Provider's own TLS Certificate. For more certificate structure options, see Microsoft Teams Direct Routing documentation.

The Microsoft Phone System Direct Routing Interface allows ***only*** TLS connections from SBCs for SIP traffic with a certificate signed by one of the Trusted Certification Authorities.

Currently, supported Certification Authorities can be found in the following link:

https://docs.microsoft.com/en-us/microsoftteams/direct-routing-plan#public-trusted-certificate-for-the-sbc

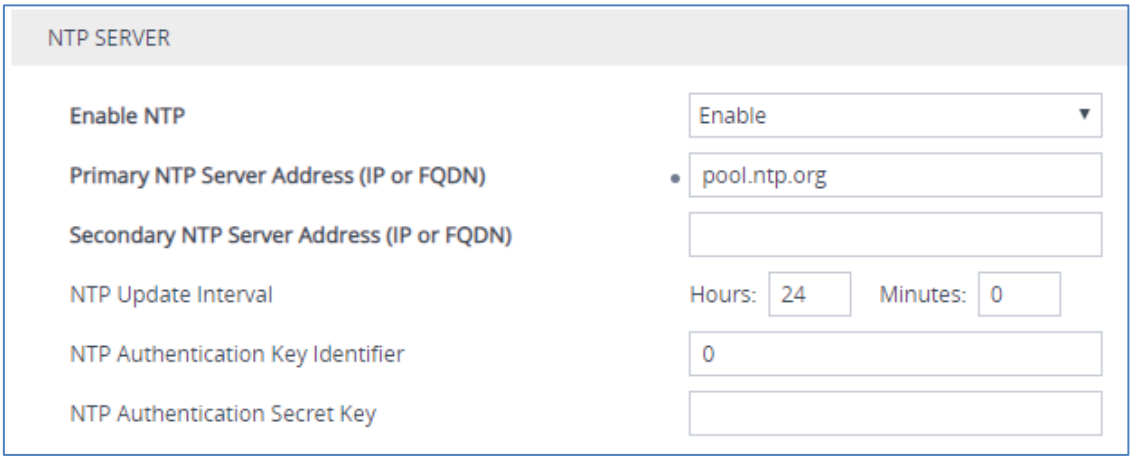
## 4.3.1 Configure the NTP Server Address

This step describes how to configure the NTP server's IP address. It is recommended to implement an NTP server (Microsoft NTP server or another global server) to ensure that the SBC receives the current date and time. This is necessary for validating certificates of remote parties.

➢ **To configure the NTP server address:**

1. Open the Time & Date page (**Setup** menu > **Administration** tab > **Time & Date**).
2. In the 'Primary NTP Server Address' field, enter the IP address of the NTP server (e.g., **pool.ntp.org**).

**Figure 4-5: Configuring NTP Server Address**




3. Click **Apply**.

### 4.3.2 Create a TLS Context for Microsoft Teams Direct Routing

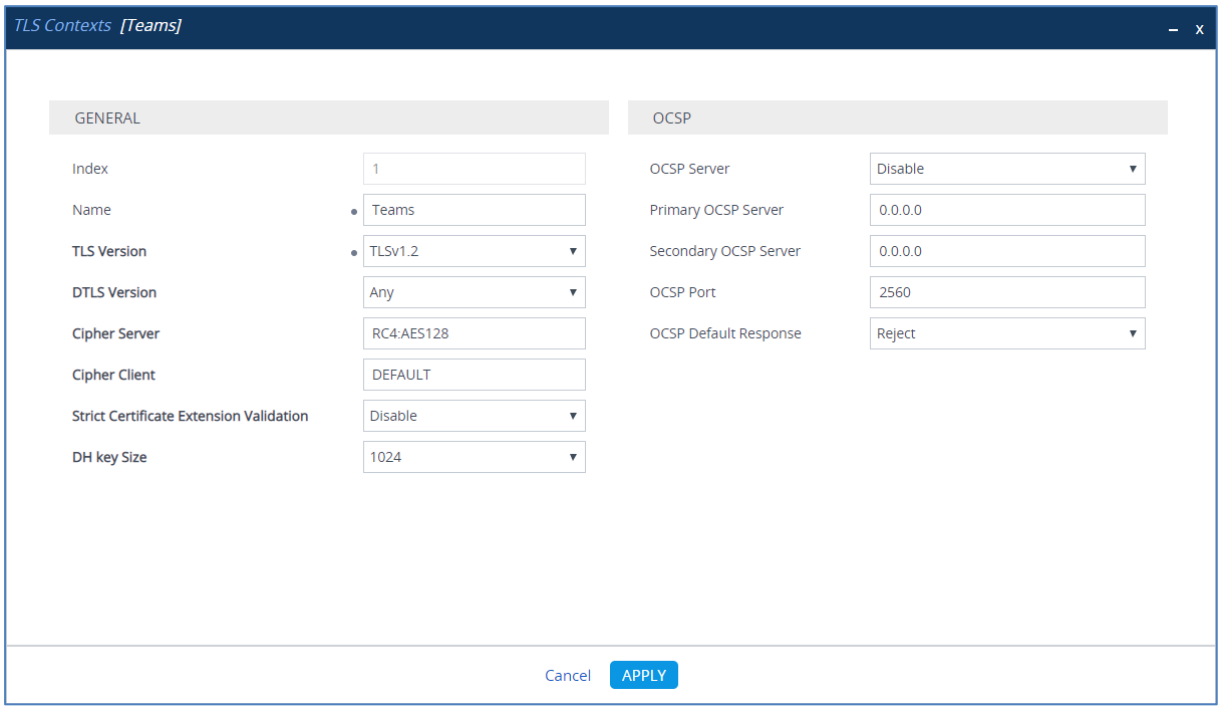
This step describes how to configure TLS Context in the SBC. AudioCodes recommends implementing only TLS version 1.2 to avoid flaws in SSL.

➢ **To configure the TLS version:**

1. Open the TLS Contexts table (**Setup** menu > **IP Network** tab > **Security** folder > **TLS Contexts**).
2. Create a new TLS Context by clicking **New** at the top of the interface, and then configure the parameters using the table below as reference:

| Parameter | Value |
|---|---|
| Index | **1** |
| Name | **Teams** (arbitrary descriptive name) |
| TLS Version | **TLSv1.2** |
| All other parameters leave unchanged at their default values | |

**Figure 4-6: Configuring TLS Context for Teams Direct Routing**




3. Click **Apply**.

### 4.3.3 Configure a Certificate

This step describes how to request a certificate for the SBC and to configure it based on the example of DigiCert Global Root CA. The certificate is used by the SBC to authenticate the connection with Microsoft Teams Direct Routing.

The procedure involves the following main steps:

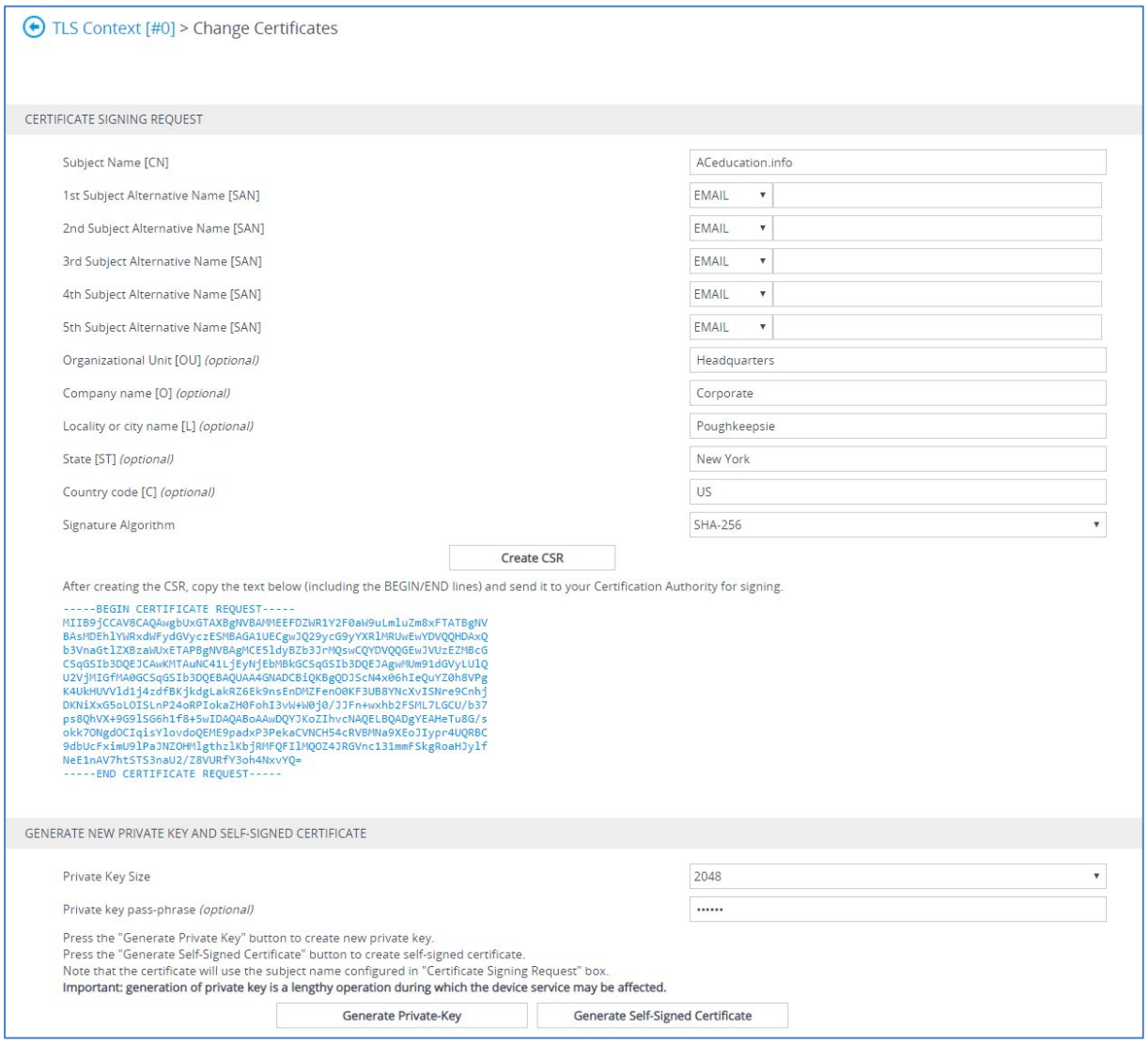
**a.** Generating a Certificate Signing Request (CSR).
**b.** Requesting Device Certificate from CA.
**c.** Obtaining Trusted Root/ Intermediate Certificate from CA.
**d.** Deploying Device and Trusted Root/ Intermediate Certificates on SBC.

**Note:** The domain portion of the Common Name [CN] must match the SIP suffix configured for Office 365 users.

➢ **To configure a certificate:**

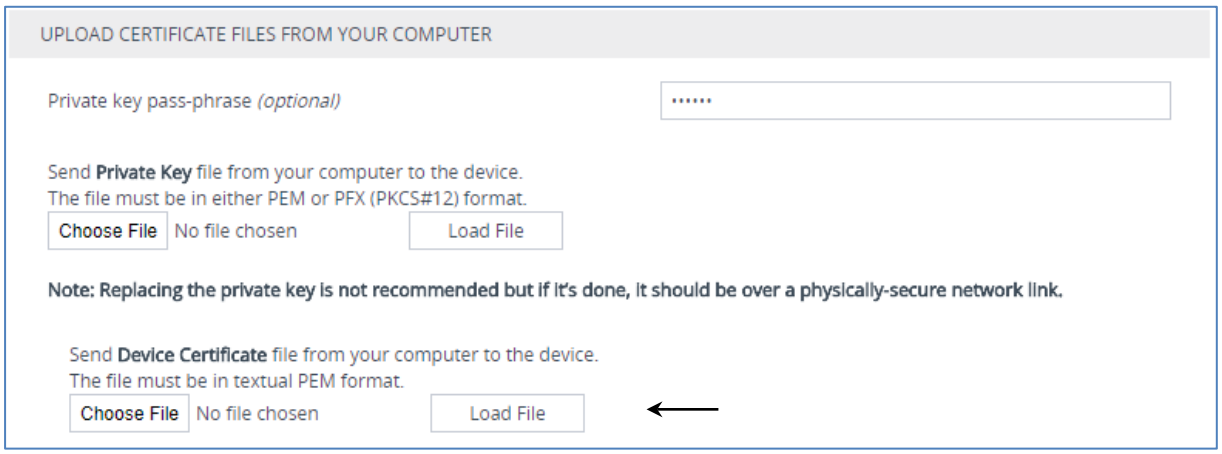
1. Open the TLS Contexts page (**Setup** menu > **IP Network** tab > **Security** folder > **TLS Contexts**).
2. In the TLS Contexts page, select the required TLS Context index row, and then click the **Change Certificate** link located below the table; the Context Certificates page appears.
3. Under the **Certificate Signing Request** group, do the following:
   a. In the 'Subject Name [CN]' field, enter the SBC FQDN name (based on example above, **ACeducation.info**).
   a. Change the 'Private Key Size' based on the requirements of your Certification Authority. Many CAs do not support private key of size 1024. In this case, you must change the key size to 2048.
   b. To change the key size on TLS Context, go to: **Generate New Private Key and Self-Signed Certificate**, change the 'Private Key Size' to **2048** and then click **Generate Private-Key**. To use **1024** as a Private Key Size value, you can click **Generate Private-Key** without changing the default key size value.
   c. Fill in the rest of the request fields according to your security provider's instructions.
   d. Click the **Create CSR** button; a textual certificate signing request is displayed in the area below the button:

**Figure 4-7: Example of Certificate Signing Request – Creating CSR**



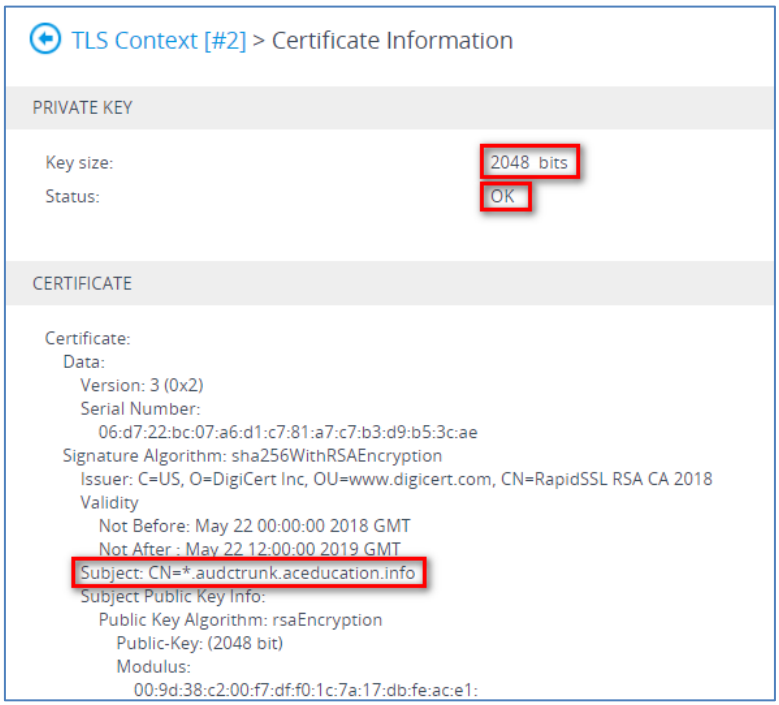

4. Copy the CSR from the line **"----BEGIN CERTIFICATE" to "END CERTIFICATE REQUEST----"** to a text file (such as Notepad), and then save it to a folder on your computer with the file name, for example *certreq.txt*.
5. Send *certreq.txt* file to the Certified Authority Administrator for signing.
6. After obtaining an SBC signed and Trusted Root/Intermediate Certificate from the CA, in the SBC's Web interface, return to the **TLS Contexts** page and do the following:
   a. In the TLS Contexts page, select the required TLS Context index row, and then click the **Change Certificate** link located below the table; the Context Certificates page appears.
   b. Scroll down to the **Upload certificates files from your computer** group, click the **Choose File** button corresponding to the '**Send Device Certificate**...' field, navigate to the certificate file obtained from the CA, and then click **Load File** to upload the certificate to the SBC.

**Figure 4-8: Uploading the Certificate Obtained from the Certification Authority**



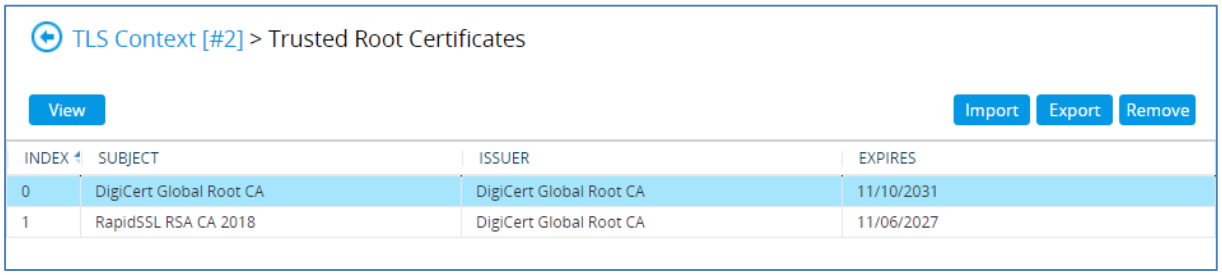

7. Confirm that the certificate was uploaded correctly. A message indicating that the certificate was uploaded successfully is displayed in blue in the lower part of the page.
8. In the SBC's Web interface, return to the **TLS Contexts** page, select the required TLS Context index row, and then click the **Certificate Information** link, located at the bottom of the TLS. Then validate the Key size, certificate status and Subject Name:

**Figure 4-9: Certificate Information Example**




9. In the SBC's Web interface, return to the **TLS Contexts** page.
   a. In the TLS Contexts page, select the required TLS Context index row, and then click the **Trusted Root Certificates** link, located at the bottom of the TLS Contexts page; the Trusted Certificates page appears.
   b. Click the **Import** button, and then select all Root/Intermediate Certificates obtained from your Certification Authority to load.
10. Click **OK**; the certificate is loaded to the device and listed in the Trusted Certificates store:

**Figure 4-10: Example of Configured Trusted Root Certificates**




11. Reset the SBC with a burn to flash for your settings to take effect.

## 4.3.4 Alternative Method of Generating and Installing the Certificate

To use the same certificate on multiple devices, you may prefer using DigiCert Certificate Utility for Windows to process the certificate request from your Certificate Authority on another machine, with this utility installed.

After you've processed the certificate request and response using the DigiCert utility, test the certificate private key and chain and then export the certificate with private key and assign a password.

➢ **To install the certificate:**

1. Open the TLS Contexts page (**Setup** menu > **IP Network** tab > **Security** folder > **TLS Contexts**).
2. In the TLS Contexts page, select the required TLS Context index row, and then click the **Change Certificate** link located below the table; the Context Certificates page appears.
3. Scroll down to the **Upload certificates files from your computer** group and do the following:
   a. Enter the password assigned during export with the DigiCert utility in the **'Private key pass-phrase'** field.
   b. Click the **Choose File** button corresponding to the 'Send **Private Key**...' field and then select the SBC certificate file exported from the DigiCert utility.

## 4.3.5 Deploy Baltimore Trusted Root Certificate

The DNS name of the Microsoft Teams Direct Routing interface is **sip.pstnhub.microsoft.com**. In this interface, a certificate is presented which is signed by Baltimore Cyber Baltimore CyberTrust Root with Serial Number: 02 00 00 b9 and SHA fingerprint: d4:de:20:d0:5e:66:fc: 53:fe:1a:50:88:2c:78:db:28:52:ca:e4:74.

To trust this certificate, your SBC *must* have the certificate in Trusted Certificates storage. Download the certificate from https://cacert.omniroot.com/bc2025.pem and follow the steps above to import the certificate to the Trusted Root storage.

**Note:** Before importing the Baltimore Root Certificate into AudioCodes' SBC, make sure it's in .PEM or .PFX format. If it isn't, you need to convert it to .PEM or .PFX format. Otherwise, you will receive a 'Failed to load new certificate' error message. To convert to PEM format, use the Windows local store on any Windows OS and then export it as 'Base-64 encoded X.509 (.CER) certificate'.

# 4.4 Configure Media Realms

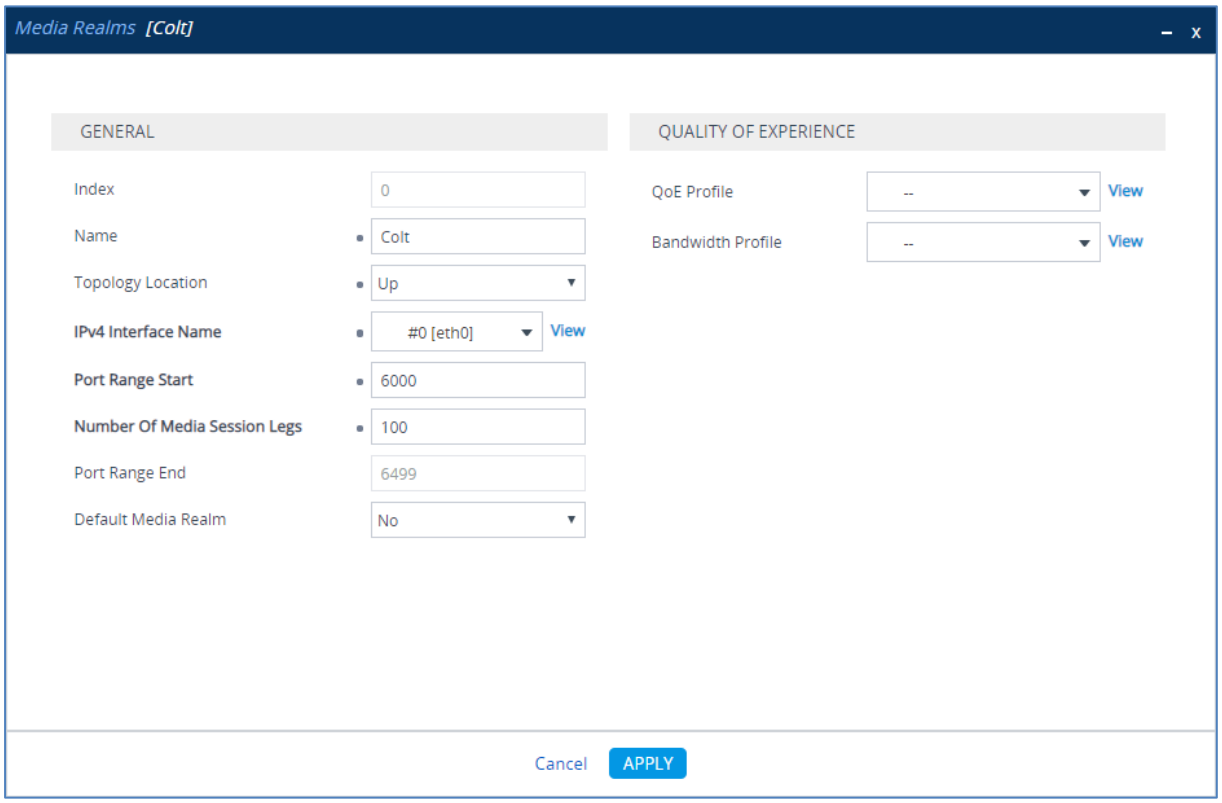
This step describes how to configure Media Realms. The simplest configuration is to create two Media Realms - one for internal (LAN) traffic and one for external (WAN) traffic.

➢ **To configure Media Realms:**

1. Open the Media Realms table (**Setup** menu > **Signaling & Media** tab > **Core Entities** folder > **Media Realms**).
2. Add a Media Realm for the LAN interface. You can use the default Media Realm (Index 0), but modify it as shown below:

| Parameter | Value |
|---|---|
| Index | **0** |
| Name | **Colt** (descriptive name) |
| IPv4 Interface Name | **eth0** |
| Port Range Start | **6000** (represents lowest UDP port number used for media on LAN) |
| Number of Media Session Legs | **100** (media sessions assigned with port range) |

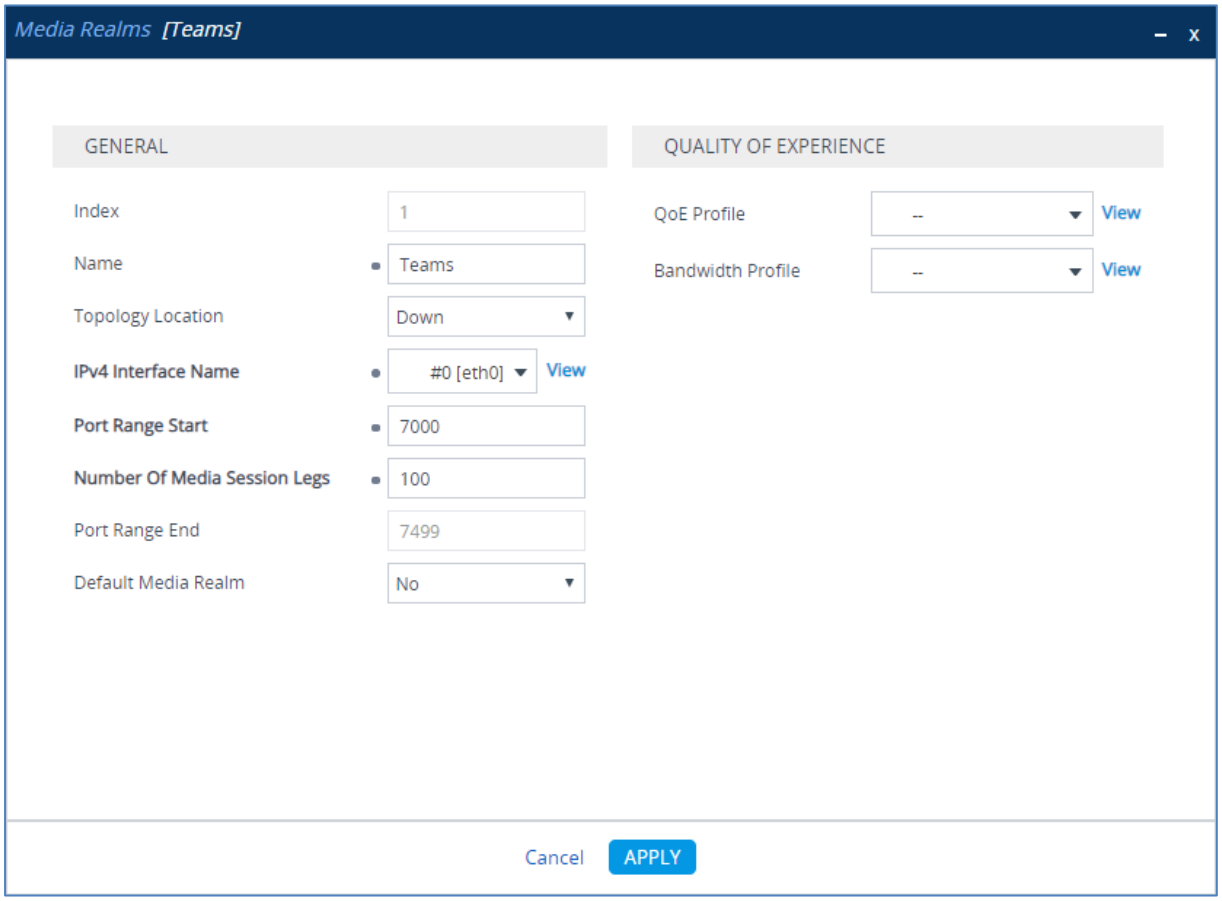
**Figure 4-11: Configuring Media Realm for SIP Trunk**



3. Configure a Media Realm for WAN traffic:

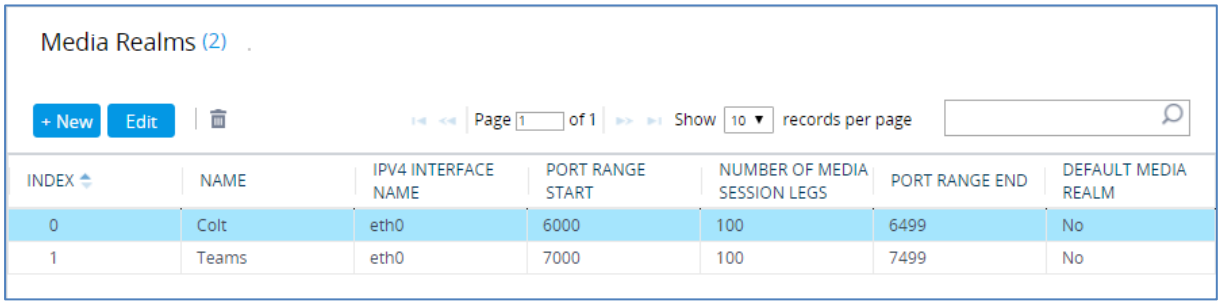
| Parameter | Value |
|---|---|
| Index | **1** |
| Name | **Teams** (arbitrary name) |
| Topology Location | **Up** |
| IPv4 Interface Name | **eth0** |
| Port Range Start | **7000** (represents lowest UDP port number used for media on WAN) |
| Number of Media Session Legs | **100** (media sessions assigned with port range) |

**Figure 4-12: Configuring Media Realm for Teams**



The configured Media Realms are shown in the figure below:

**Figure 4-13: Configured Media Realms in Media Realm Table**



## 4.5 Configure SIP Signaling Interfaces

This step describes how to configure SIP Interfaces. For the interoperability test topology, two SIP Interfaces must be configured for the SBC, one towards the SIP Trunk and another one towards the Microsoft Teams Direct Routing Interface.

➢ **To configure SIP Interfaces:**

1. Open the SIP Interfaces table (**Setup** menu > **Signaling & Media** tab > **Core Entities** folder > **SIP Interfaces**).
2. Add a SIP Interface for the LAN interface. You can use the default SIP Interface (Index 0), but modify it as shown below:

| Parameter | Value |
|---|---|
| Index | **0** |
| Name | **Colt** (arbitrary descriptive name) |
| Network Interface | **eth0** |
| Application Type | **SBC** |
| UDP Port | **5060** (according to Service Provider requirement) |
| TCP and TLS Port | **0** |
| Media Realm | **Colt** |

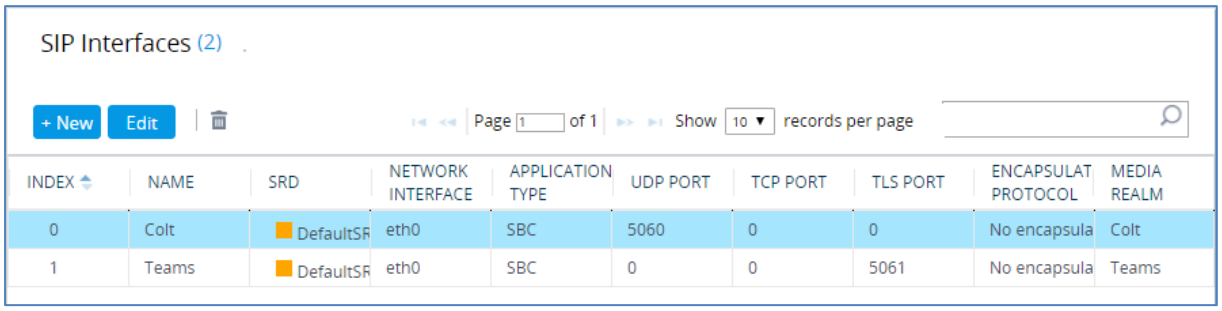
**Note:** The Direct Routing interface can only use TLS transport for a SIP call. It does not SIP TCP support due to security reasons. The SIP port may be any port of your choice. When pairing the SBC with Office 365, the chosen port is specified in the pairing command.

3. Configure a SIP Interface for the WAN:

| Parameter | Value |
|---|---|
| Index | **1** |
| Name | **Teams** (arbitrary descriptive name) |
| Network Interface | **eth0** |
| Application Type | **SBC** |
| UDP and TCP Port | **0** |
| TLS Port | **5061** (as configured in the Office 365) |
| Enable TCP Keepalive | **Enable** |
| Classification Failure Response Type | **0** |
| Media Realm | **Teams** |

The configured SIP Interfaces are shown in the figure below:

**Figure 4-14: Configured SIP Interfaces in SIP Interface Table**



# 4.6 Configure Proxy Sets

This step describes how to configure Proxy Sets. The Proxy Set defines the destination address (IP address or FQDN) of the IP entity server. Proxy Sets can also be used to configure load balancing between multiple servers.

For the interoperability test topology, two Proxy Sets need to be configured for the following IP entities:

- Colt SIP Trunk
- Microsoft Teams Direct Routing

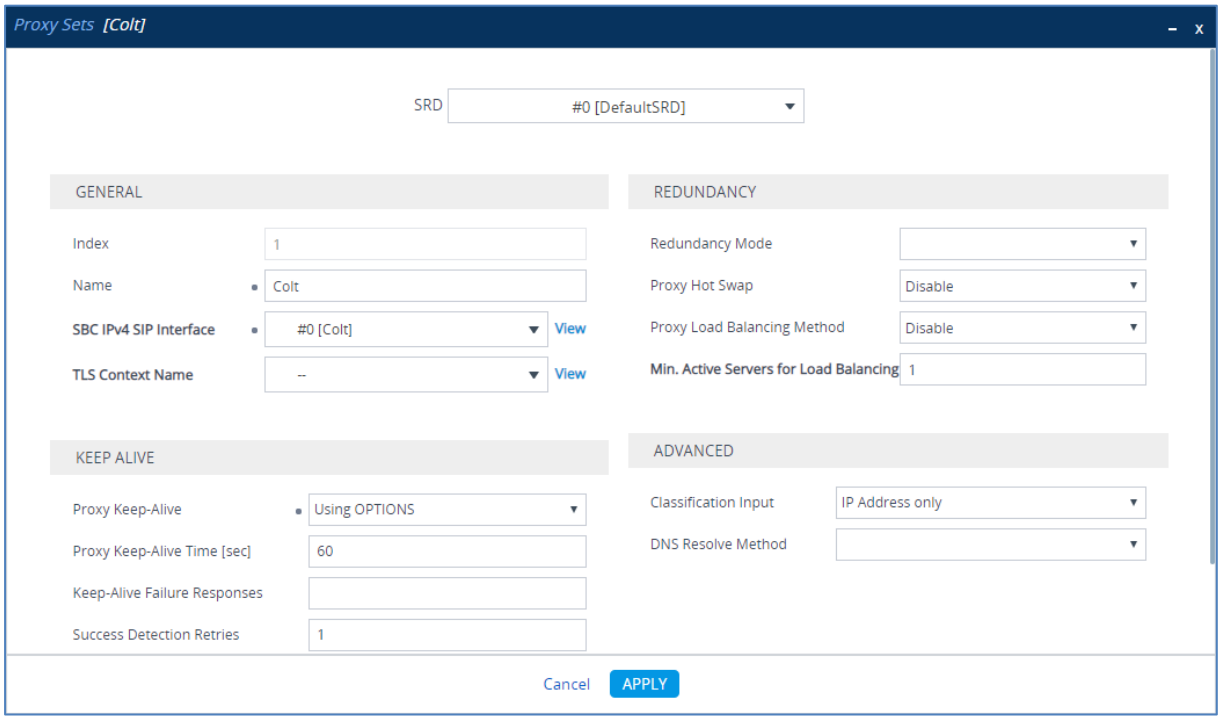
The Proxy Sets will later be applied to the VoIP network by assigning them to IP Groups.

➢ **To configure Proxy Sets:**

1. Open the Proxy Sets table (**Setup** menu > **Signaling & Media** tab > **Core Entities** folder > **Proxy Sets**).
2. Add a Proxy Set for the Colt SIP Trunk:

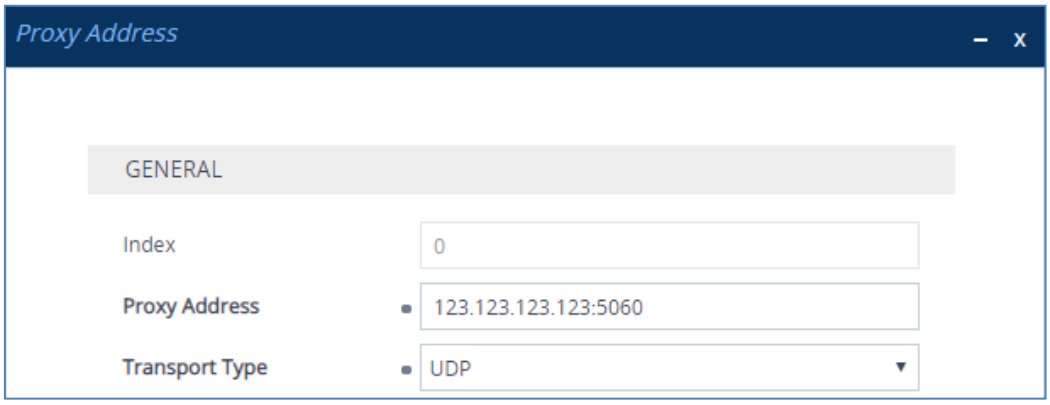
| Parameter | Value |
|---|---|
| Index | **1** |
| Name | **Colt** |
| SBC IPv4 SIP Interface | **Colt** |
| Proxy Keep-Alive | **Using Options** |

**Figure 4-15: Configuring Proxy Set for Colt SIP Trunk**




a. Select the index row of the Proxy Set that you added, and then click the **Proxy Address** link located below the table; the Proxy Address table opens.
b. Click **New**; the following dialog box appears:

**Figure 4-16: Configuring Proxy Address for Colt SIP Trunk**




**c.** Configure the address of the Proxy Set according to the parameters described in the table below.

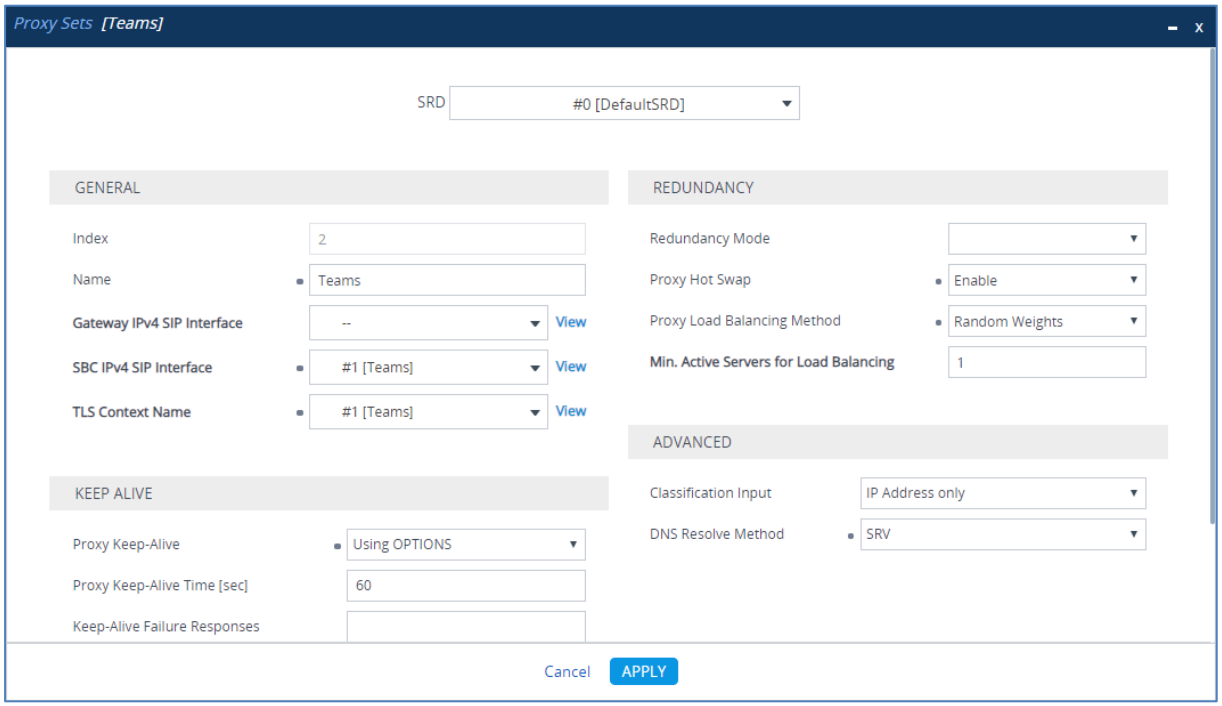
| Parameter | Value |
|---|---|
| Index | **0** |
| Proxy Address | **123.123.123.123:5060** (IP address / FQDN and destination port) |
| Transport Type | **UDP** |

**d.** Click **Apply**.

**3.** Add a Proxy Set for the Microsoft Teams Direct Routing as shown below:

| Parameter | Value |
|---|---|
| Index | **2** |
| Name | **Teams** (arbitrary descriptive name) |
| SBC IPv4 SIP Interface | **Teams** |
| TLS Context Name | **Teams** |
| Proxy Keep-Alive | **Using Options** |
| Proxy Hot Swap | **Enable** |
| Proxy Load Balancing Method | **Random Weights** |
| DNS Resolve Method | **SRV** |

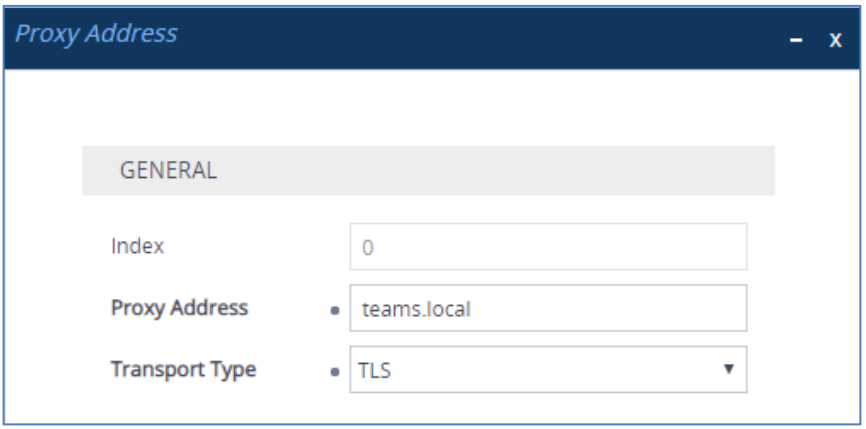
**Figure 4-17: Configuring Proxy Set for Microsoft Teams Direct Routing**




a. Select the index row of the Proxy Set that you added, and then click the **Proxy Address** link located below the table; the Proxy Address table opens.

b. Click **New**; the following dialog box appears:

**Figure 4-18: Configuring Proxy Address for Microsoft Teams Direct Routing Interface**



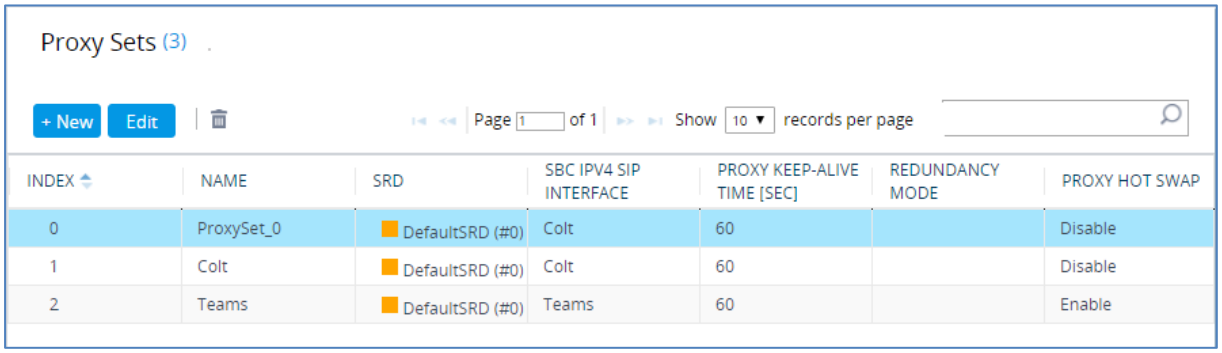

c. Configure the address of the Proxy Set according to the parameters described in the table below.

| Parameter | Value |
|---|---|
| Index | **0** |
| Proxy Address | **teams.local** (Teams Direct Routing FQDN) |
| Transport Type | **TLS** |

d. Click **Apply**.

The configured Proxy Sets are shown in the figure below:

**Figure 4-19: Configured Proxy Sets in Proxy Sets Table**



## 4.7 Configure the Internal SRV Table

The Internal SRV table resolves host names to DNS A-Records. Three different A-Records can be assigned to each host name, where each A-Record contains the host name, priority, weight, and port.

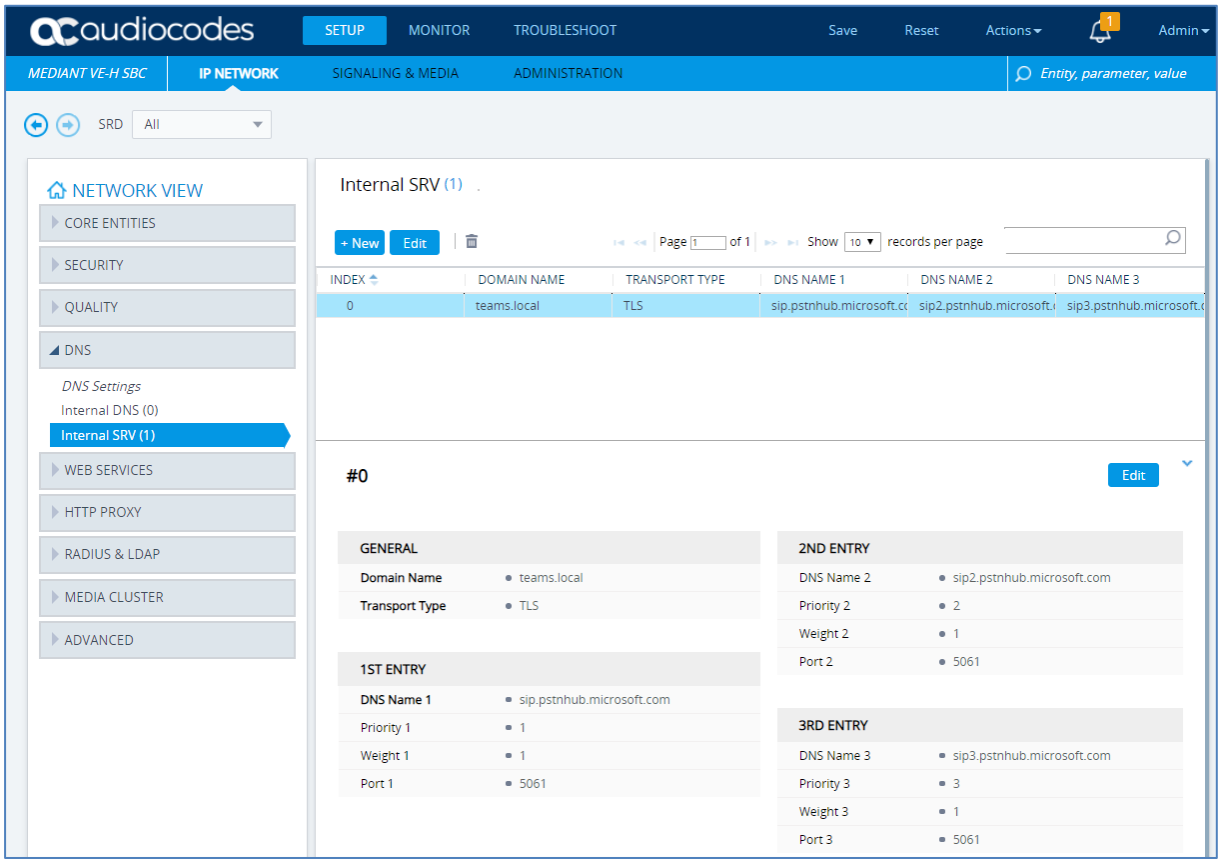
➢ **To configure the Internal SRV Table:**

1. Open the Internal SRV table (**Setup** menu > **IP Network** tab > **DNS** folder > **Internal SRV**).
2. Click **New** to add the SRV record for **teams.local** and use the table below as configuration reference.

**Table 4-1: Configuration Example of the Internal SRV Table**

| Parameter | Value |
|---|---|
| Domain Name | **teams.local** (FQDN is case-sensitive; configure in line with the configuration of the Teams Proxy Set) |
| Transport Type | **TLS** |
| **1st ENTRY** | |
| DNS Name 1 | **sip.pstnhub.microsoft.com** |
| Priority 1 | **1** |
| Weight 1 | **1** |
| Port 1 | **5061** |
| **2nd ENTRY** | |
| DNS Name 2 | **sip2.pstnhub.microsoft.com** |
| Priority 2 | **2** |
| Weight 2 | **1** |
| Port 2 | **5061** |
| **3rd ENTRY** | |
| DNS Name 3 | **sip3.pstnhub.microsoft.com** |
| Priority 3 | **3** |
| Weight 3 | **1** |
| Port 3 | **5061** |

**Figure 4-20: Example of the Internal SRV Table**



# 4.8 Configure Coders

This step describes how to configure coders (termed *Coder Group*). As Microsoft Teams Direct Routing supports the SILK and OPUS coders while the network connection to Colt SIP Trunk may restrict operation with a dedicated coders list, you need to add a Coder Group with the supported coders for each leg, the Microsoft Teams Direct Routing and the Colt SIP Trunk.

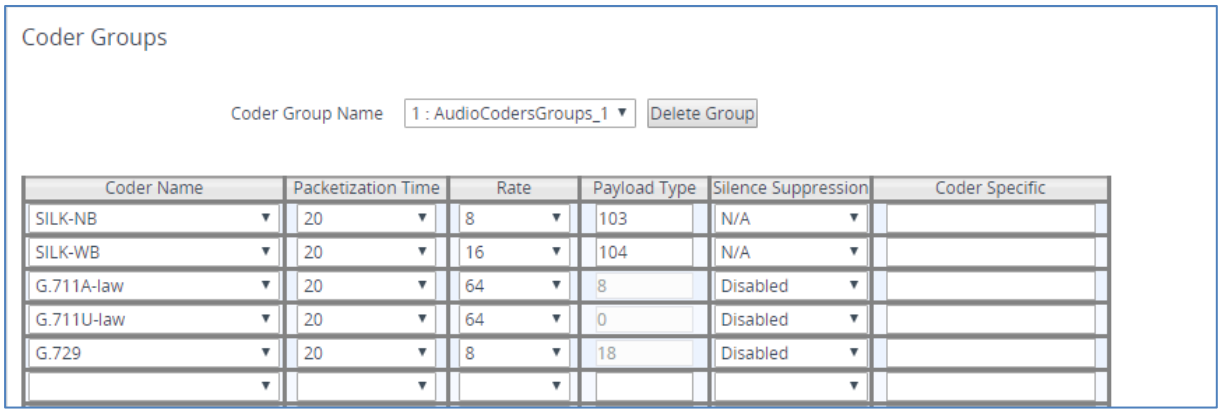
Note that the Coder Group ID for this entity will be assigned to its corresponding IP Profile in the next step.

➢ **To configure coders:**

1. Open the Coder Groups table (**Setup** menu > **Signaling & Media** tab > **Coders & Profiles** folder > **Coder Groups**).
2. Configure a Coder Group for Microsoft Teams Direct Routing:

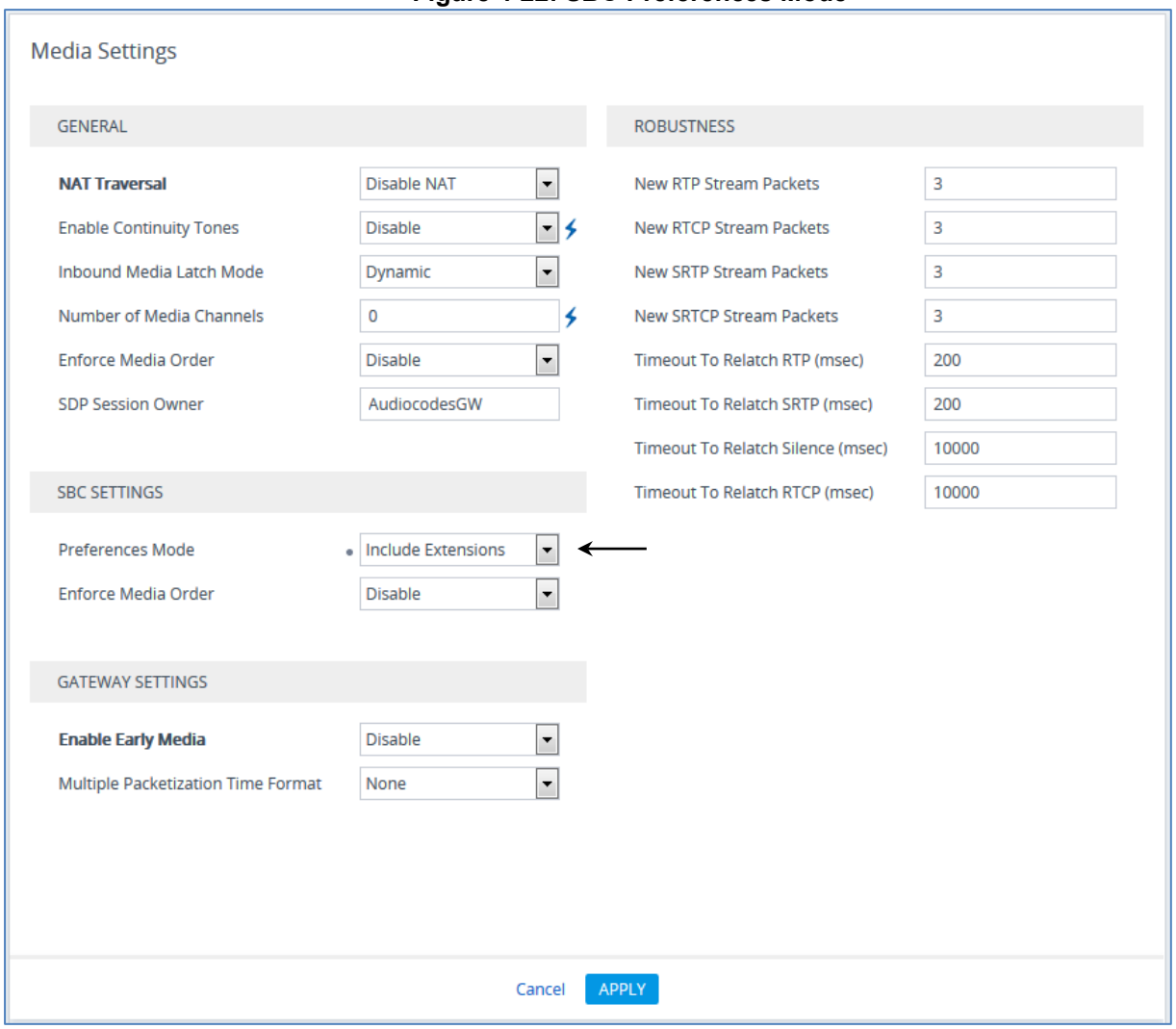
| Parameter | Value |
|---|---|
| Coder Group Name | **AudioCodersGroups_1** |
| Coder Name | ▪ **SILK-NB**<br>▪ **SILK-WB**<br>▪ **G.711 A-law**<br>▪ **G.711 U-law**<br>▪ **G.729** |

**Figure 4-21: Configuring Coder Group for Microsoft Teams Direct Routing**



3. Open the Media Settings page (**Setup** menu > **Signaling & Media** tab > **Media** folder > **Media Settings**).

**Figure 4-22: SBC Preferences Mode**




4. From the '**Preferences Mode**' drop-down list, select **Include Extensions**.
5. Click **Apply**.

# 4.9 Configure IP Profiles

This step describes how to configure IP Profiles. The IP Profile defines a set of call capabilities relating to signaling (e.g., SIP message terminations such as REFER) and media (e.g., coder and transcoding method).

In this interoperability test topology, IP Profiles need to be configured for the following IP entities:

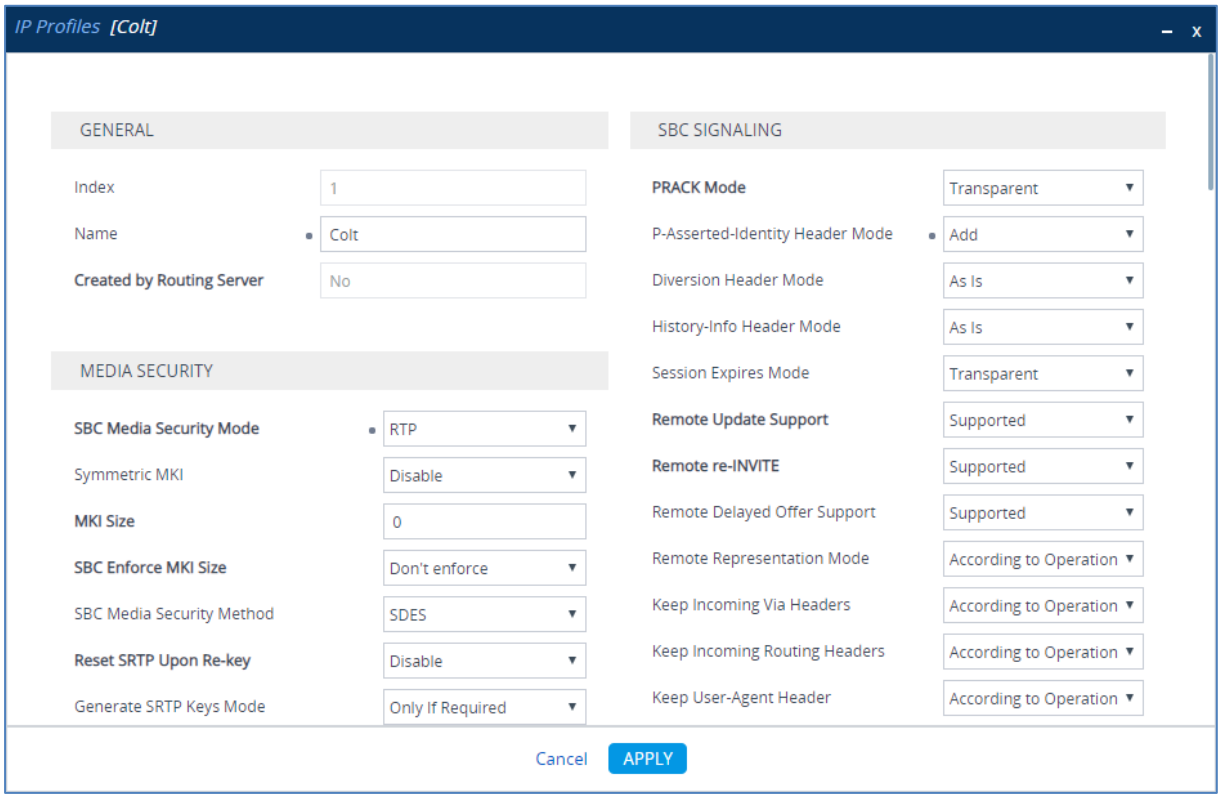
- Colt SIP trunk – to operate in non-secure mode using RTP and SIP over UDP
- Microsoft Teams Direct Routing – to operate in secure mode using SRTP and SIP over TLS

➢ **To configure an IP Profile for the Colt SIP Trunk:**

1. Open the IP Profiles table (**Setup** menu > **Signaling & Media** tab > **Coders & Profiles** folder > **IP Profiles**).
2. Click **New**, and then configure the parameters as follows:

| Parameter | Value |
|---|---|
| **General** | |
| Index | **1** |
| Name | **Colt** |
| **Media Security** | |
| SBC Media Security Mode | **RTP** |
| **SBC Signaling** | |
| P-Asserted-Identity Header Mode | **Add** (required for anonymous calls) |
| **SBC Forward and Transfer** | |
| Remote REFER Mode | **Handle Locally** |
| Remote Replaces Mode | **Handle Locally** |
| Plat RBT To Transferee | **Yes** |
| Remote 3xx Mode | **Handle Locally** |

**Figure 4-23: Configuring IP Profile for Colt SIP Trunk**




3. Click **Apply**.

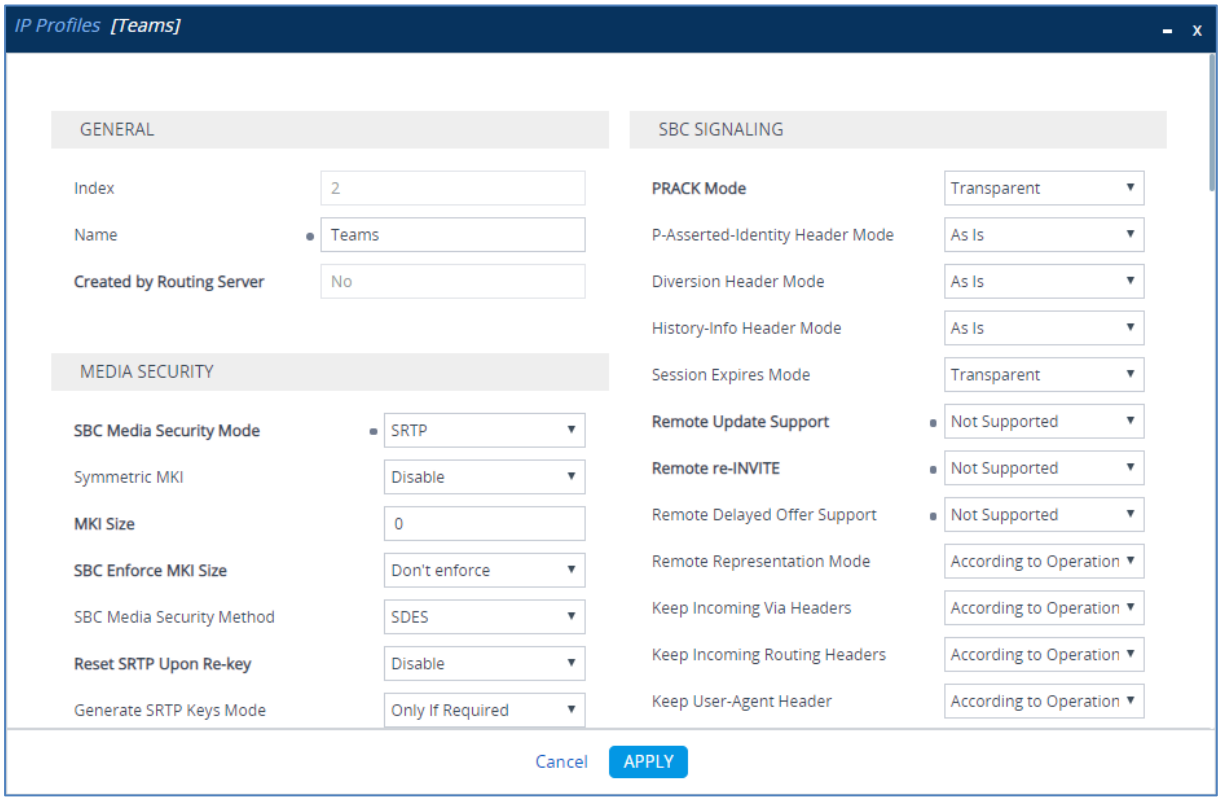
### ➢ To configure IP Profile for the Microsoft Teams Direct Routing:

1. Open the IP Profiles table (**Setup** menu > **Signaling & Media** tab > **Coders & Profiles** folder > **IP Profiles**).
2. Click **New**, and then configure the parameters as follows:

| Parameter | Value |
|---|---|
| **General** | |
| Index | **2** |
| Name | **Teams** (arbitrary descriptive name) |
| **Media Security** | |
| SBC Media Security Mode | **SRTP** |
| **SBC Early Media** | |
| Remote Early Media RTP Detection Mode | **By Media** (required, as Microsoft Teams Direct Routing does not send RTP immediately to remote side when it sends a SIP 18x response) |
| **SBC Media** | |
| Extension Coders Group | **AudioCodersGroups_1** |
| RTCP Mode | **Generate Always** (required for Hold and Mute scenarios, when Colt SIP Trunk does not send RTCP packets, however, the Microsoft Teams |

| | |
|---|---|
| | drops the calls if the RTCP packets are not received). |
| ICE Mode | **Lite** (required only when Media Bypass enabled on Microsoft Teams) |
| **SBC Signaling** | |
| Remote Update Support | **Not Supported** |
| Remote re-INVITE Support | **Not Supported** |
| Remote Delayed Offer Support | **Not Supported** |
| **SBC Forward and Transfer** | |
| Remote REFER Mode | **Handle Locally** |
| Remote 3xx Mode | **Handle Locally** |
| **SBC Hold** | |
| Remote Hold Format | **Inactive** (some SIP Trunk may answer with a=inactive and IP=0.0.0.0 in response to the Re-Invite with Hold request from Teams. Microsoft Media Stack doesn't support this format. So, SBC will replace 0.0.0.0 with its IP address) |

**Figure 4-24: Configuring IP Profile for Microsoft Teams Direct Routing**




3. Click **Apply**.

## 4.10 Configure IP Groups

This step describes how to configure IP Groups. The IP Group represents an IP entity on the network with which the SBC communicates. This can be a server (e.g., IP PBX or ITSP) or it can be a group of users (e.g., LAN IP phones). For servers, the IP Group is typically used to define the server's IP address by associating it with a Proxy Set. Once IP Groups are configured, they are used to configure IP-to-IP routing rules for denoting source and destination of the call.

In this interoperability test topology, IP Groups must be configured for the following IP entities:

- Colt SIP Trunk
- Teams Direct Routing

➢ **To configure IP Groups:**

1. Open the IP Groups table (**Setup** menu > **Signaling & Media** tab > **Core Entities** folder > **IP Groups**).
2. Configure an IP Group for the Colt SIP Trunk:

| Parameter | Value |
|---|---|
| Index | **1** |
| Name | **Colt** |
| Type | **Server** |
| Proxy Set | **Colt** |
| IP Profile | **Colt** |
| Media Realm | **Colt** |
| SIP Group Name | (according to ITSP requirement) |

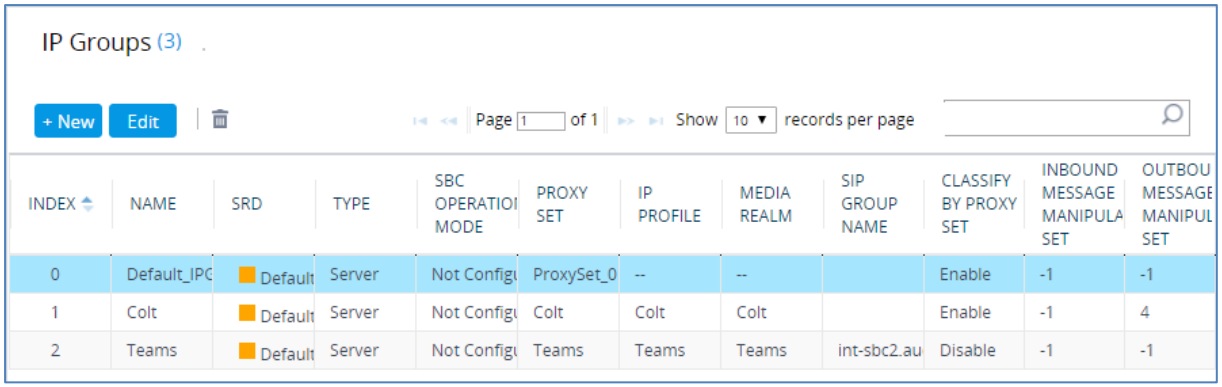
3. Configure an IP Group for the Microsoft Teams Direct Routing:

| Parameter | Value |
|---|---|
| Index | **2** |
| Name | **Teams** |
| Topology Location | **Up** |
| Type | **Server** |
| Proxy Set | **Teams** |
| IP Profile | **Teams** |
| Media Realm | **Teams** |
| Classify By Proxy Set | **Disable** |
| SIP Group Name | **< FQDN name of your SBC in the Microsoft Teams tenant >**<br>(For example, sbc1.customers.ACeducation.info) |
| Local Host Name | **< FQDN name of your SBC in the Microsoft Teams tenant >**<br>(For example, sbc1.customers.ACeducation.info) |
| Always Use Src Address | **Yes** |

| | |
|---|---|
| DTLS Context | **Teams** |
| Proxy Keep-Alive using IP Group settings | **Enable** |

The configured IP Groups are shown in the figure below:

**Figure 4-25: Configured IP Groups in IP Group Table**



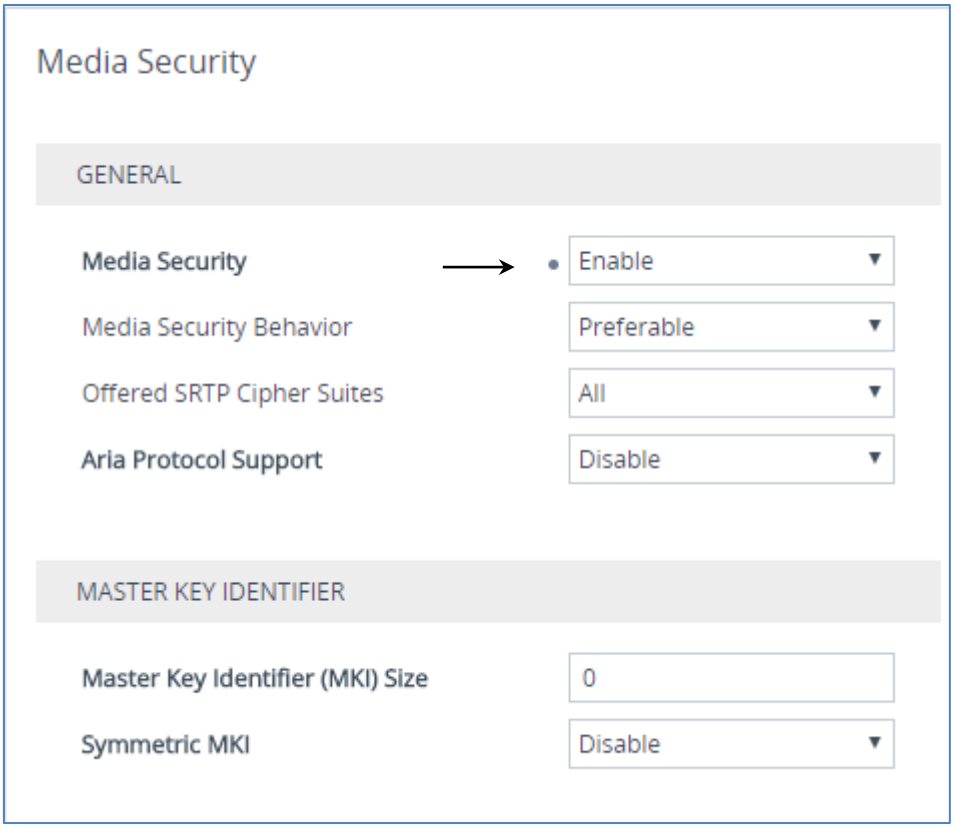
# 4.11 Configure SRTP

This step describes how to configure media security. The Direct Routing Interface needs to use of SRTP only, so you need to configure the SBC to operate in the same manner.

➢ **To configure media security:**

1. Open the Media Security page (**Setup menu > Signaling & Media tab > Media folder > Media Security**).

**Figure 4-26: Configuring SRTP**




2. From the 'Media Security' drop-down list, select **Enable** to enable SRTP.
3. Click **Apply**.

# 4.12 Configuring Message Condition Rules

This step describes how to configure the Message Condition Rules. A Message Condition defines special conditions (pre-requisites) for incoming SIP messages. These rules can be used as additional matching criteria for the IP-to-IP routing rules in the IP-to-IP Routing table.

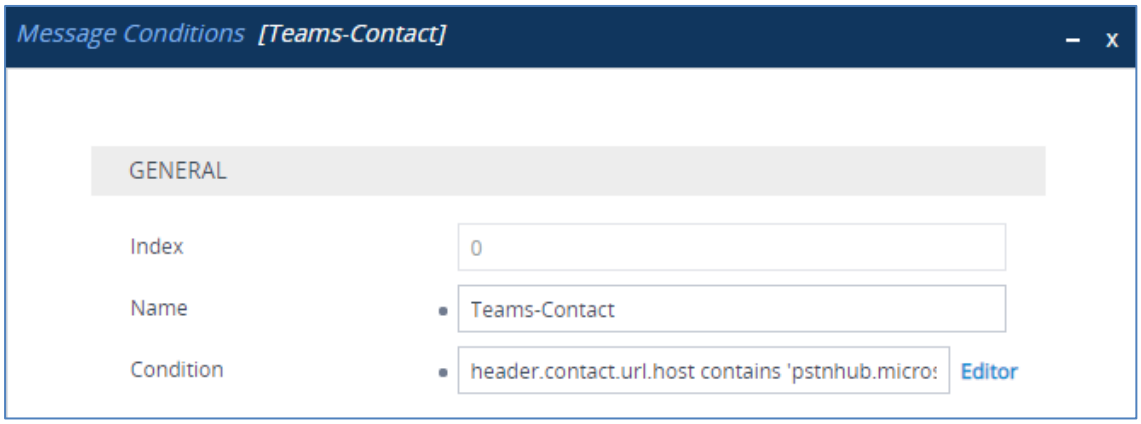
The following condition verifies that the Contact header contains Microsoft Teams FQDN.

➢ **To configure a Message Condition rule:**

1. Open the Message Conditions table (**Setup** menu > **Signaling & Media** tab > **Message Manipulation** folder > **Message Conditions**).
2. Click **New**, and then configure the parameters as follows:

| Parameter | Value |
|---|---|
| Index | **0** |
| Name | **Teams-Contact** (arbitrary descriptive name) |
| Condition | **header.contact.url.host contains 'pstnhub.microsoft.com'** |

**Figure 4-27: Configuring Condition Table**




3. Click **Apply**.

# 4.13 Configuring Classification Rules

This step describes how to configure Classification rules. A Classification rule classifies incoming SIP dialog-initiating requests (e.g., INVITE messages) to a 'source' IP Group. The source IP Group is the SIP entity that sent the SIP dialog request. Once classified, the device uses the IP Group to process the call (manipulation and routing).

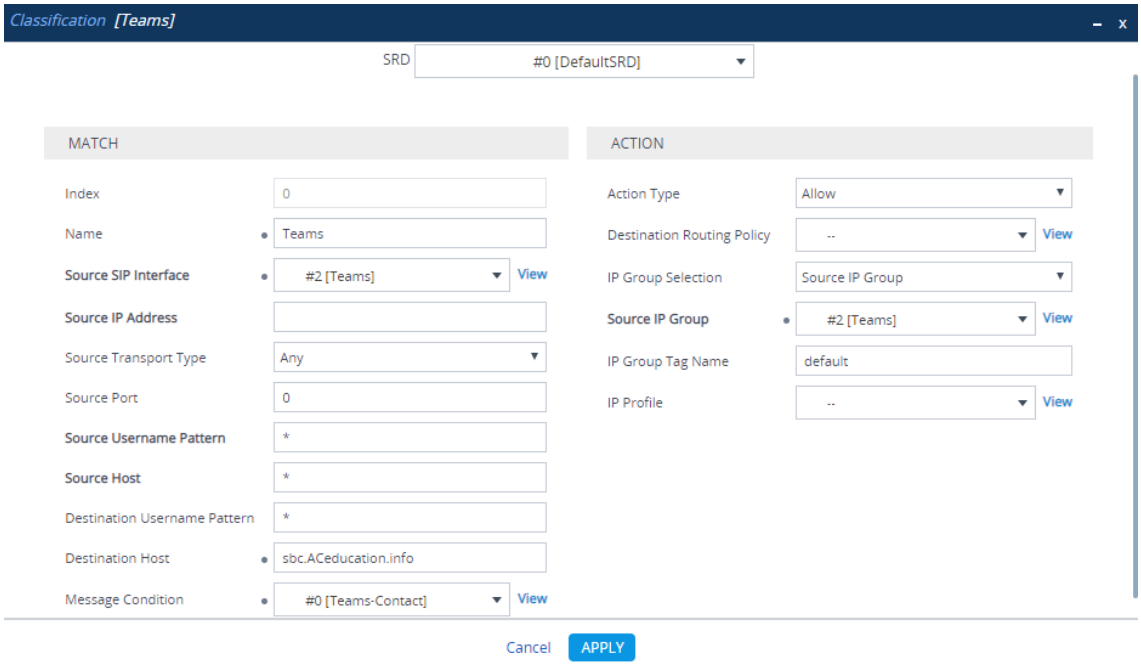
You can also use the Classification table for employing SIP-level access control for successfully classified calls, by configuring Classification rules with whitelist and blacklist settings. If a Classification rule is configured as a whitelist ("Allow"), the device accepts the SIP dialog and processes the call. If the Classification rule is configured as a blacklist ("Deny"), the device rejects the SIP dialog.

➢ **To configure a Classification rule:**

1. Open the Classification table (**Setup** menu > **Signaling & Media** tab > **SBC** folder > **Classification Table**).
2. Click **New**, and then configure the parameters as follows:

| Parameter | Value |
|---|---|
| Index | **0** |
| Name | **Teams** |
| Source SIP Interface | **Teams** |
| Destination Host | **sbc.ACeducation.info** (FQDN name of your SBC in the Microsoft Teams tenant) |
| Message Condition | **Teams-Contact** |
| Action Type | **Allow** |
| Source IP Group | **Teams** |

**Figure 4-28: Configuring Classification Rule**




3. Click **Apply**.

## 4.14 Configure IP-to-IP Call Routing Rules

This step describes how to configure IP-to-IP call routing rules. These rules define the routes for forwarding SIP messages (e.g., INVITE) received from one IP entity to another. The SBC selects the rule whose configured input characteristics (e.g., IP Group) match those of the incoming SIP message. If the input characteristics do not match the first rule in the table, they are compared to the second rule, and so on, until a matching rule is located. If no rule is matched, the message is rejected. The routing rules use the configured IP Groups (as configured in Section 4.10 on page 35,) to denote the source and destination of the call.

For the interoperability test topology, the following IP-to-IP routing rules need to be configured to route calls between Teams Direct Routing and Colt SIP Trunk:

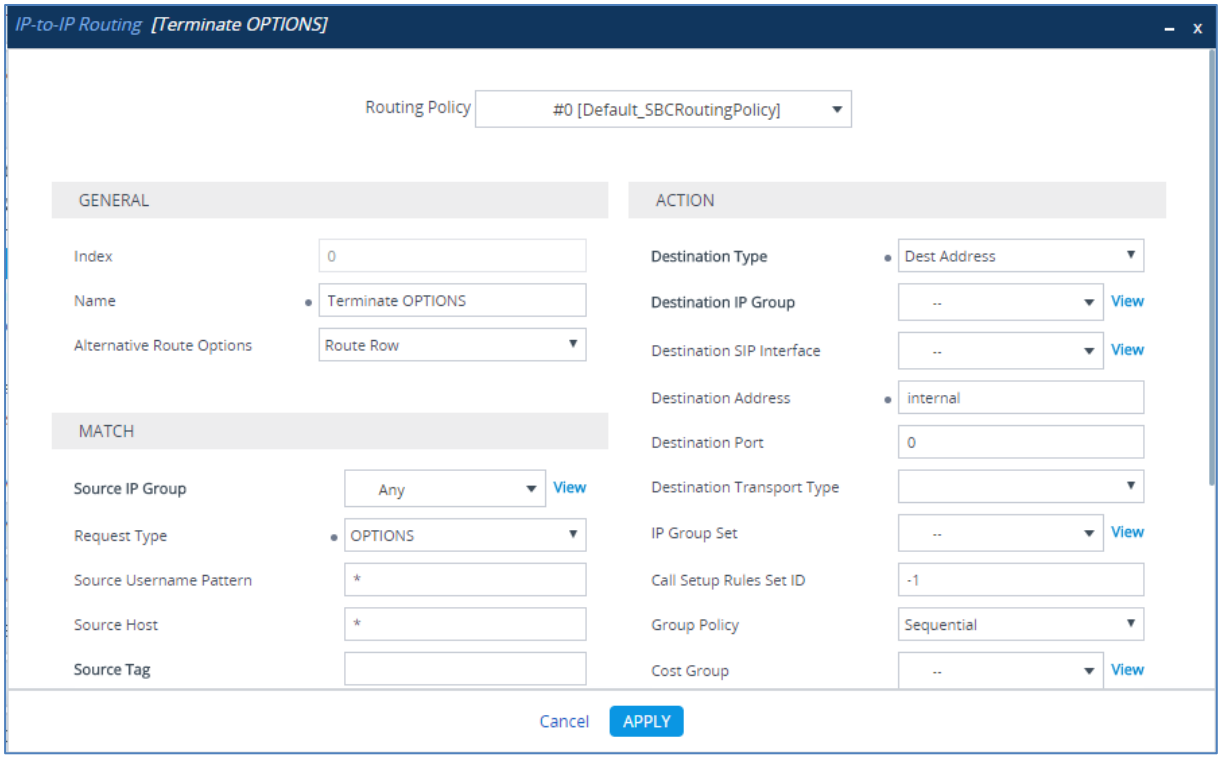
- Terminate SIP OPTIONS messages on the SBC that are received from any entity
- Terminate REFER messages to Teams Direct Routing
- Calls from Teams Direct Routing to Colt SIP Trunk
- Calls from Colt SIP Trunk to Teams Direct Routing

➢ **To configure IP-to-IP routing rules:**

1. Open the IP-to-IP Routing table (**Setup** menu > **Signaling & Media** tab > **SBC** folder > **Routing** > **IP-to-IP Routing**).
2. Configure a rule to terminate SIP OPTIONS messages:
   a. Click **New**, and then configure the parameters as follows:

| Parameter | Value |
|---|---|
| Index | **0** |
| Name | **Terminate OPTIONS** (arbitrary descriptive name) |
| Source IP Group | **Any** |
| Request Type | **OPTIONS** |
| Destination Type | **Dest Address** |
| Destination Address | **internal** |

**Figure 4-29: Configuring IP-to-IP Routing Rule for Terminating SIP OPTIONS**

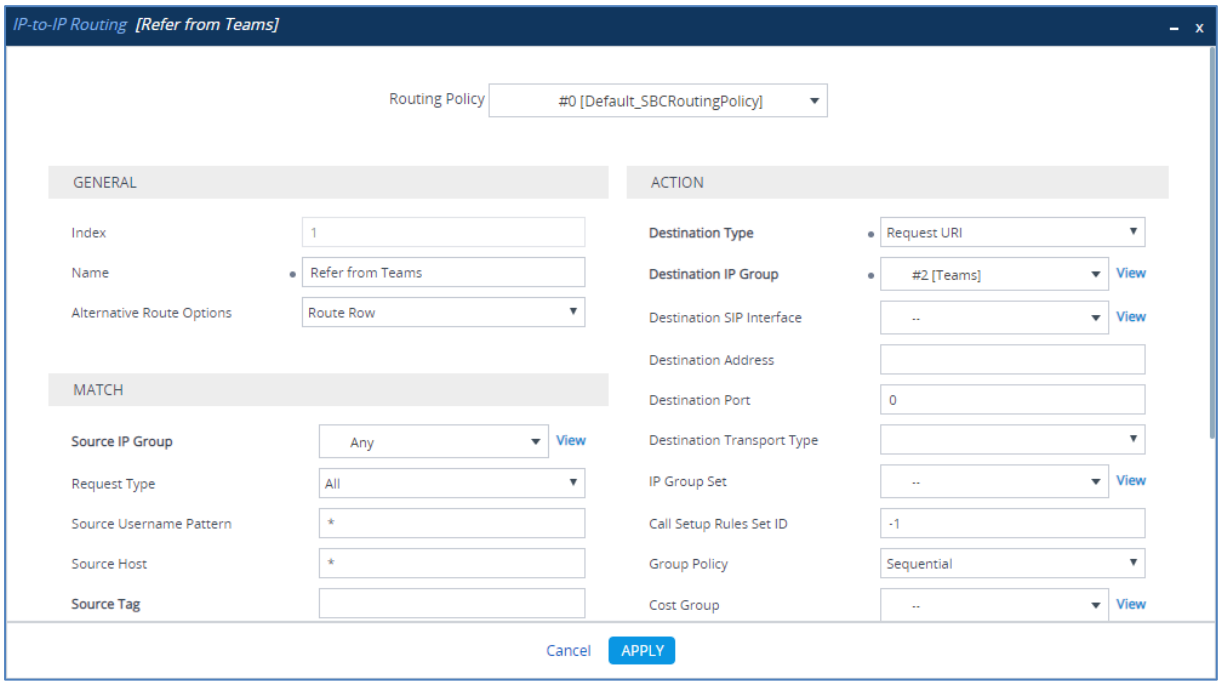



   b. Click **Apply**.

3. Configure a rule to terminate REFER messages to Teams Direct Routing:
   a. Click **New**, and then configure the parameters as follows:

| Parameter | Value |
|---|---|
| Index | **1** |
| Route Name | **Refer from Teams** (arbitrary descriptive name) |
| Source IP Group | **Any** |
| Call Triger | **REFER** |
| ReRoute IP Group | **Teams** |
| Destination Type | **Request URI** |
| Destination IP Group | **Teams** |

**Figure 4-30: Configuring IP-to-IP Routing Rule for REFER from Teams**

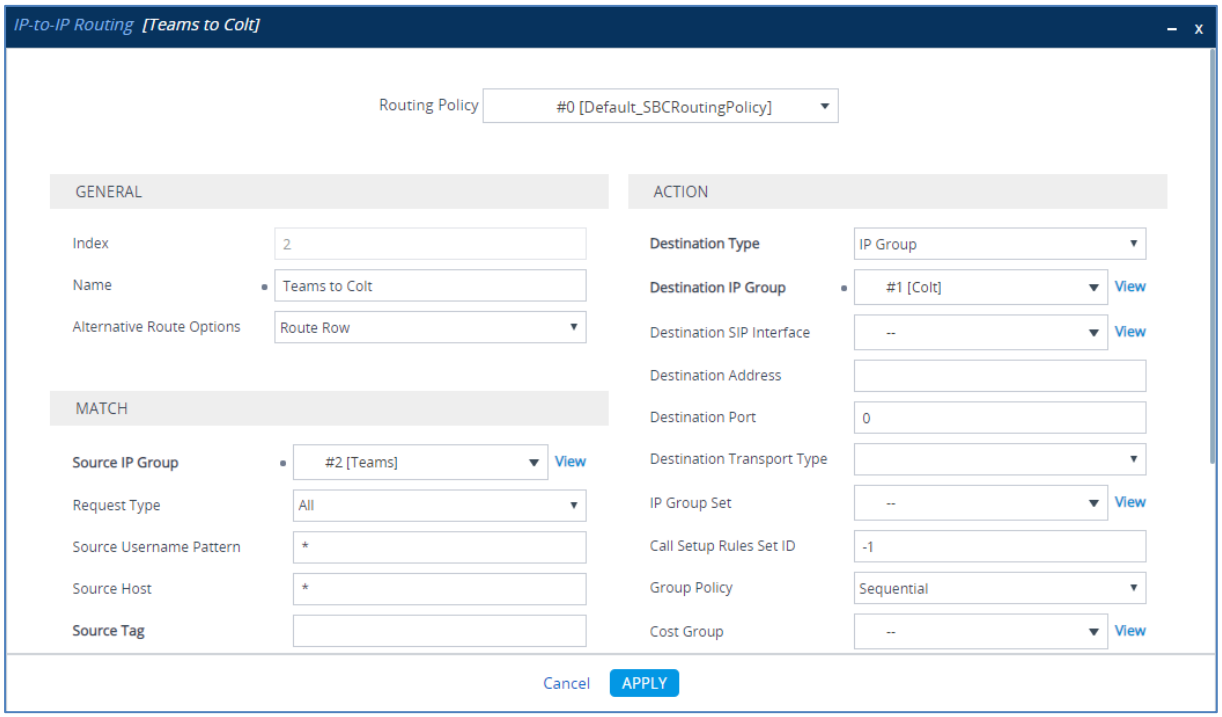



   b. Click **Apply**.

4. Configure a rule to route calls from Teams Direct Routing to Colt SIP Trunk:
   a. Click **New**, and then configure the parameters as follows:

| Parameter | Value |
|---|---|
| Index | **2** |
| Route Name | **Teams to Colt** (arbitrary descriptive name) |
| Source IP Group | **Teams** |
| Destination Type | **IP Group** |
| Destination IP Group | **Colt** |

**Figure 4-31: Configuring IP-to-IP Routing Rule for Teams to SIP Trunk**

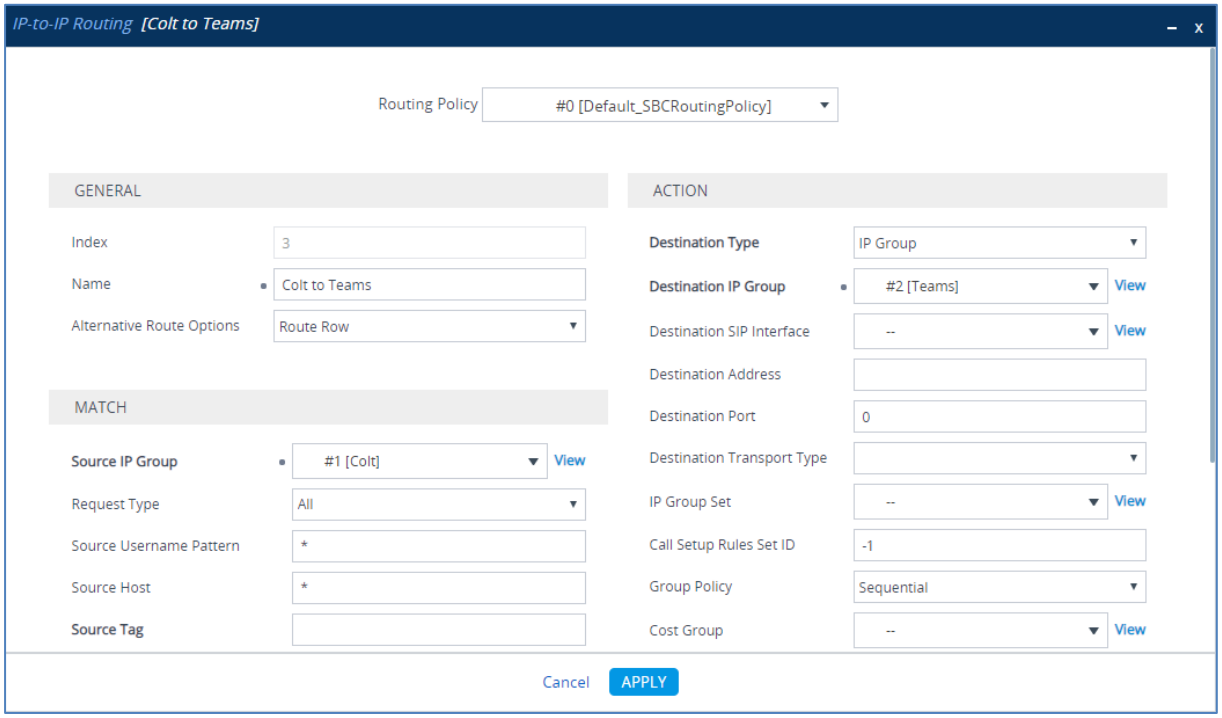



   b. Click **Apply**.

5. Configure rule to route calls from Colt SIP Trunk to Teams Direct Routing:
   a. Click **New**, and then configure the parameters as follows:

| Parameter | Value |
|---|---|
| Index | **3** |
| Route Name | **Colt to Teams** (arbitrary descriptive name) |
| Source IP Group | **Colt** |
| Destination Type | **IP Group** |
| Destination IP Group | **Teams** |

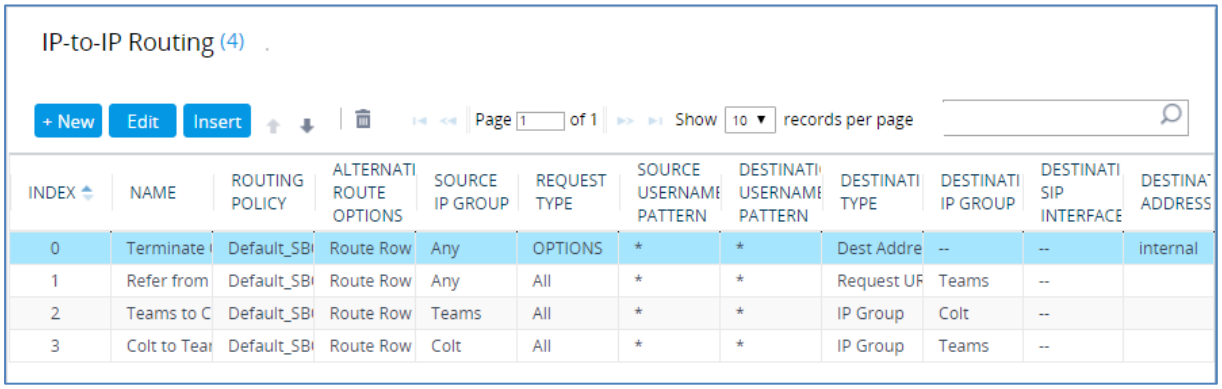
**Figure 4-32: Configuring IP-to-IP Routing Rule for SIP Trunk to Teams**




   b. Click **Apply.**

The configured routing rules are shown in the figure below:

**Figure 4-33: Configured IP-to-IP Routing Rules in IP-to-IP Routing Table**




**Note:** The routing configuration may change according to your specific deployment topology.

## 4.15 Configure Number Manipulation Rules

This step describes how to configure IP-to-IP manipulation rules. These rules manipulate the SIP Request-URI user part (source or destination number). The manipulation rules use the configured IP Groups (as configured in Section 4.10 on page 35) to denote the source and destination of the call.

**Note:** Adapt the manipulation table according to your environment dial plan.

For this interoperability test topology, no manipulation is configured because both the Colt SIP Trunk and Microsoft Teams Direct Routing use the same E.164 number format.

# 4.16 Configure Message Manipulation Rules

This step describes how to configure SIP message manipulation rules. SIP message manipulation rules can include insertion, removal, and/or modification of SIP headers. Manipulation rules are grouped into Manipulation Sets, enabling you to apply multiple rules to the same SIP message (IP entity).

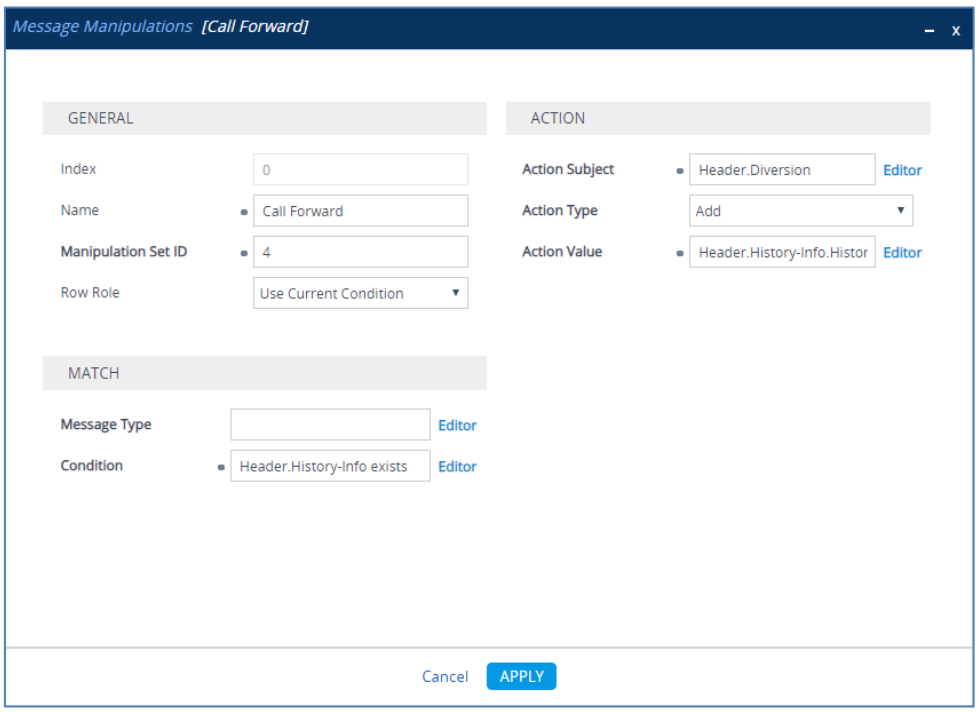
Once you have configured the SIP message manipulation rules, you need to assign them to the relevant IP Group (in the IP Group table) and determine whether they must be applied to inbound or outbound messages.

➢ **To configure SIP message manipulation rule:**

1. Open the Message Manipulations page (**Setup** menu > **Signaling & Media** tab > **Message Manipulation** folder > **Message Manipulations**).
2. Configure a new manipulation rule (Manipulation Set 4) for Colt SIP Trunk. This rule applies to messages sent to the Colt SIP Trunk IP Group in a call forwarding scenario. This rule adds the SIP Diversion header with the value from the SIP History-Info Header.

| Parameter | Value |
|---|---|
| Index | **0** |
| Name | **Call Forward** |
| Manipulation Set ID | **4** |
| Condition | **Header.History-Info exists** |
| Action Subject | **Header.Diversion** |
| Action Type | **Add** |
| Action Value | **Header.History-Info.HistoryInfo** |

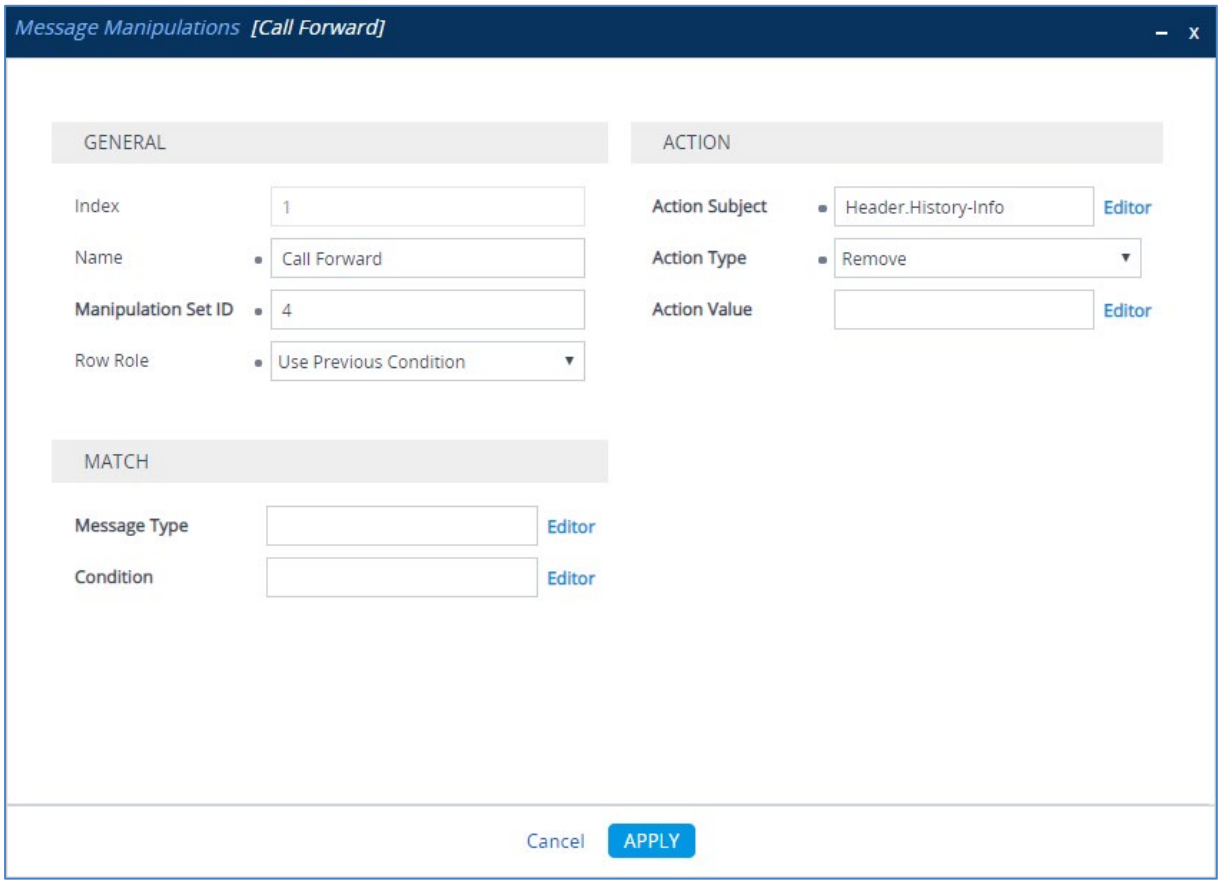
**Figure 4-34: Configuring SIP Message Manipulation Rule 0 (for Colt SIP Trunk)**



3. If the manipulation rule Index 0 (above) is executed, then the following rule is also executed. It removes the SIP History-Info header.

| Parameter | Value |
|---|---|
| Index | **1** |
| Name | **Call Forward** |
| Manipulation Set ID | **4** |
| Row Role | **Use Previous Condition** |
| Action Subject | **Header.History-Info** |
| Action Type | **Remove** |

**Figure 4-35: Configuring SIP Message Manipulation Rule 1 (for Colt SIP Trunk)**



4. Configure another manipulation rule (Manipulation Set 4) for Colt SIP Trunk. This rule is applied to response messages sent to the Colt SIP Trunk IP Group for Rejected Calls with different reason causes, initiated by the Teams Direct Routing IP Group. This replaces the reason cause code with the value '21' (Call Rejected), because Colt SIP Trunk not recognizes other causes codes and continue to try to setup call.

| Parameter | Value |
|---|---|
| Index | **2** |
| Name | **Change Reason Cause Code to 21** |
| Manipulation Set ID | **4** |
| Message Type | **Any.Response** |
| Condition | **Header.Reason exists** |
| Action Subject | **Header.Reason.Reason.Cause** |
| Action Type | **Modify** |
| Action Value | **'21'** |

**Figure 4-36: Configuring SIP Message Manipulation Rule 2 (for Colt SIP Trunk)**

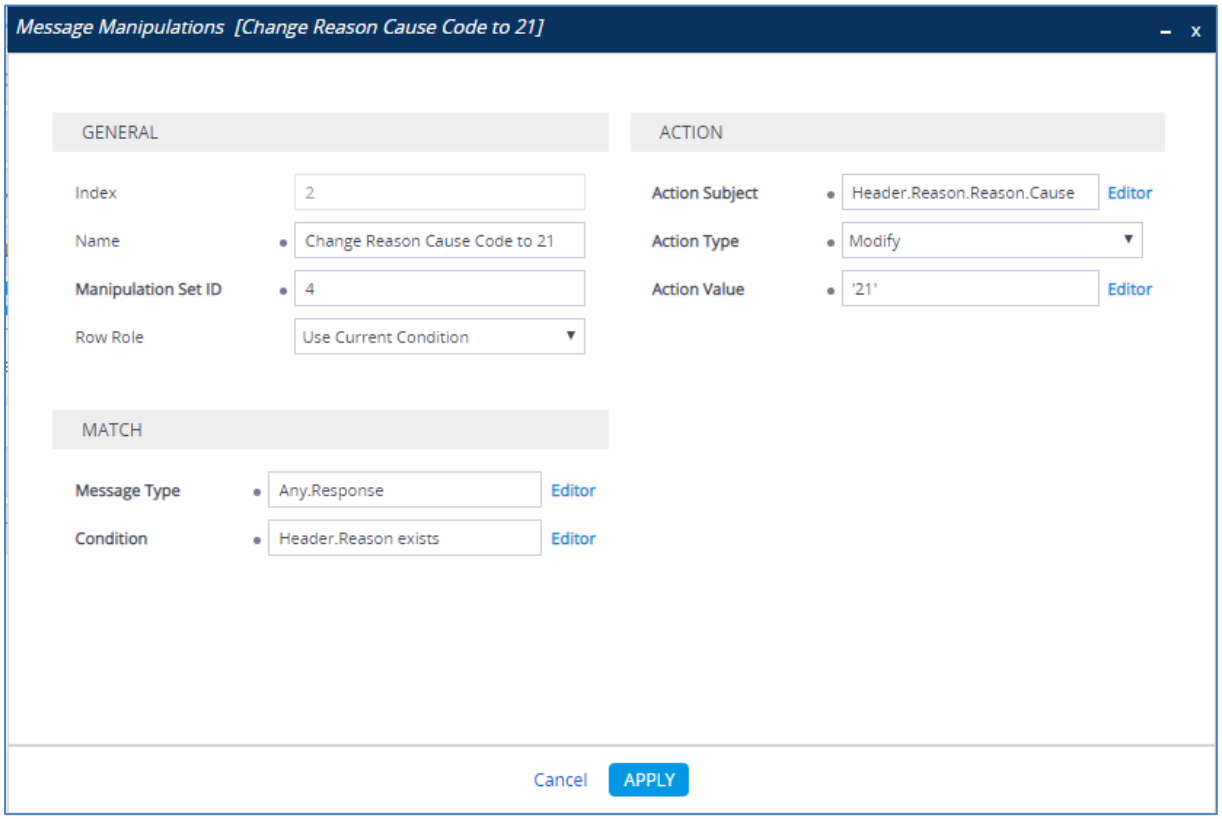
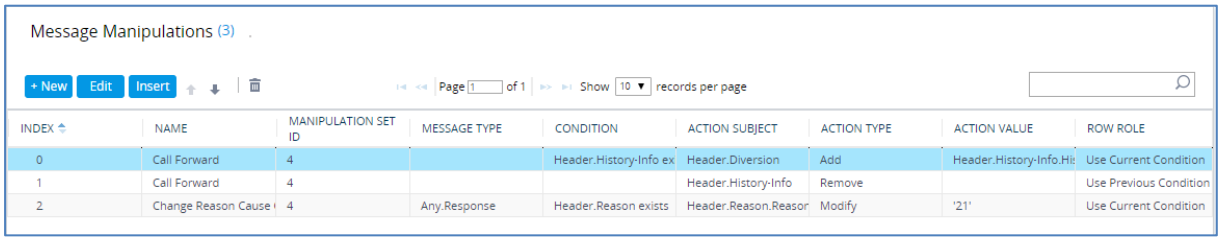


**Figure 4-37: Example of Configured SIP Message Manipulation Rules**

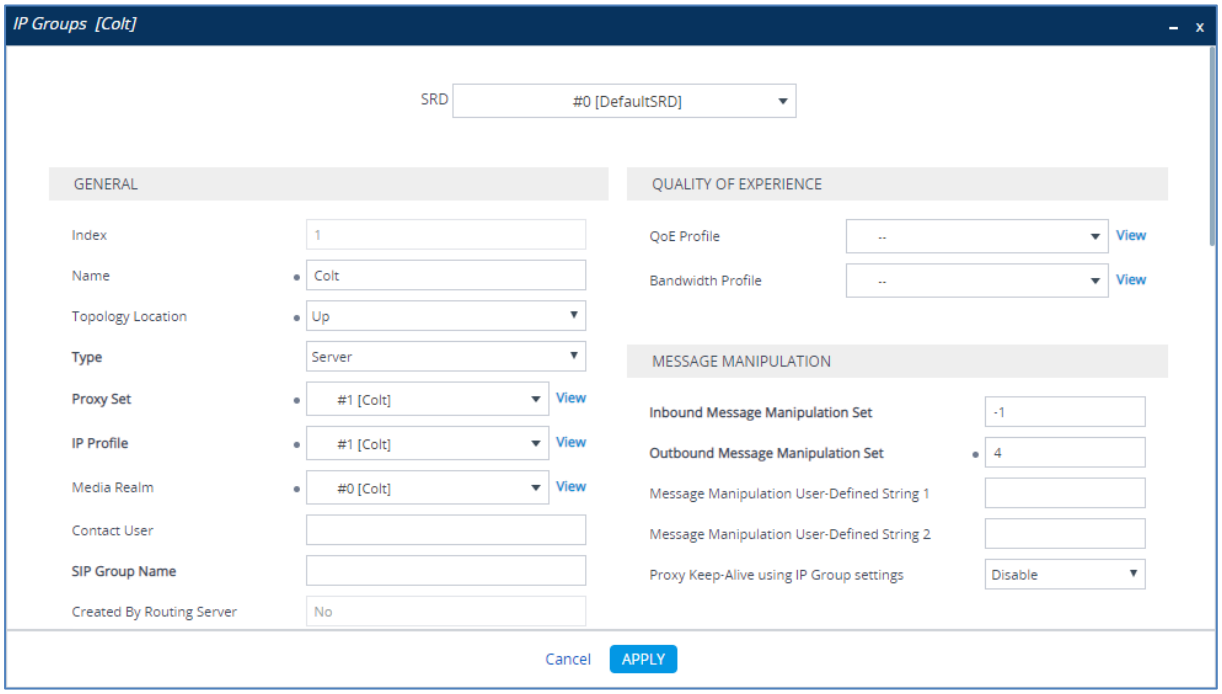



The table displayed below includes SIP message manipulation rules which are grouped together under Manipulation Set ID 4 and which are executed for messages sent to the Colt SIP Trunk IP Group. These rules are specifically required to enable proper interworking between Colt SIP Trunk and Teams Direct Routing. Refer to the *User's Manual* for further details concerning the full capabilities of header manipulation.

| Rule Index | Rule Description | Reason for Introducing Rule |
|---|---|---|
| **0** | This rule applies to messages sent to the Colt SIP Trunk IP Group in a call forwarding scenario. This rule adds the SIP Diversion header with the value from the SIP History-Info Header. | For Call Forward scenarios, Colt SIP Trunk requests SIP Diversion header instead of SIP History-Info header, sent from the Microsoft Teams. |
| **1** | If the manipulation rule Index 0 (above) is executed, then the following rule is also executed. It removes History Info Header. | |
| **2** | This rule is applied to response messages sent to the Colt SIP Trunk IP Group for Rejected Calls with different reason causes, initiated by the Teams Direct Routing IP Group. This replaces the reason cause code with the value '21' (Call Rejected). | Colt SIP Trunk does not recognize other cause codes and continues to try to setup the call. |

5. Assign Manipulation Set ID 4 to the Colt SIP trunk IP Group:
   a. Open the IP Groups table (**Setup** menu > **Signaling & Media** tab > **Core Entities** folder > **IP Groups**).
   b. Select the row of the Colt SIP trunk IP Group, and then click **Edit**.
   c. Set the 'Outbound Message Manipulation Set' field to **4**.

**Figure 4-38: Assigning Manipulation Set 4 to the Colt SIP Trunk IP Group**




   d. Click **Apply**.

# 4.17 Miscellaneous Configuration

This section describes miscellaneous SBC configuration.

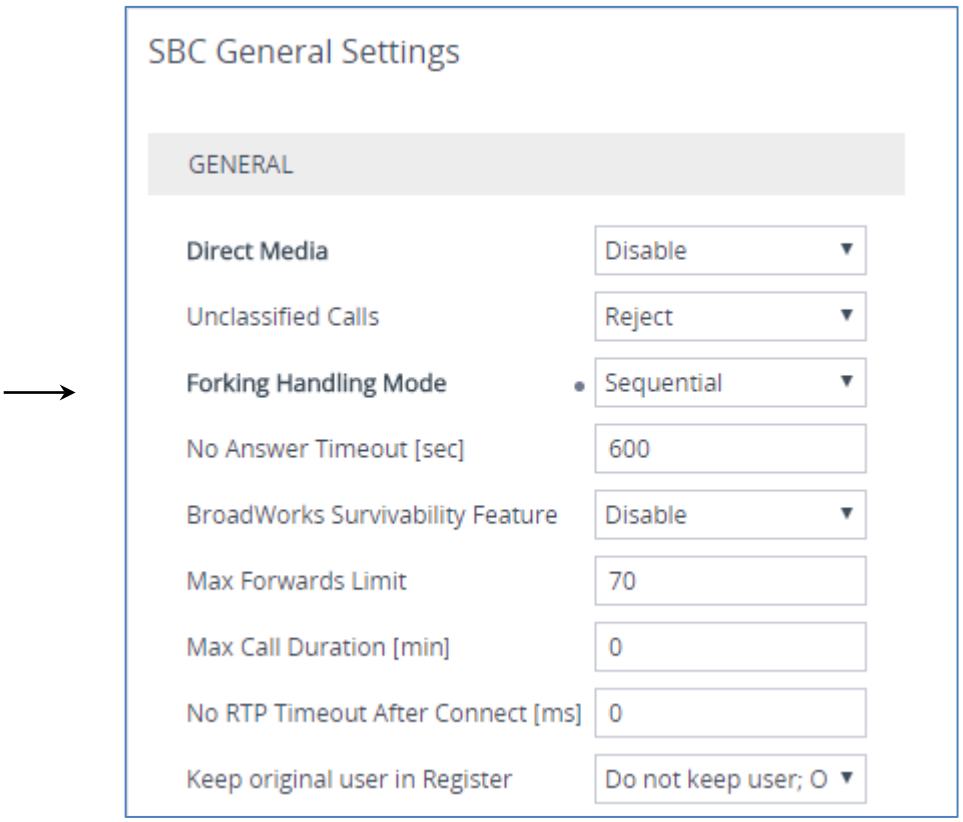
## 4.17.1 Configure Call Forking Mode

This step describes how to configure the SBC's handling of SIP 18x responses received for call forking of INVITE messages. For the interoperability test topology, if a SIP 18x response with SDP is received, the SBC opens a voice stream according to the received SDP. The SBC re-opens the stream according to subsequently received 18x responses with SDP or plays a ringback tone if a 180 response without SDP is received. It is mandatory to set this field for the Teams Direct Routing environment.

➢ **To configure call forking:**

1. Open the SBC General Settings page (**Setup** menu > **Signaling & Media** tab > **SBC** folder > **SBC General Settings**).
2. From the 'SBC Forking Handling Mode' drop-down list, select **Sequential**.

**Figure 4-39: Configuring Forking Mode**




3. Click **Apply**.

### 4.17.2 Optimizing CPU Cores Usage for a Specific Service (relevant for Mediant 9000 and Software SBC only)

This step describes how to optimize the SBC's CPU cores usage for a specified profile to achieve maximum capacity for that profile. The supported profiles include:

- SIP profile – improves SIP signaling performance, for example, SIP calls per second (CPS)
- SRTP profile – improves maximum number of SRTP sessions
- Transcoding profile – enables all DSP-required features, for example, transcoding and voice in-band detectors

➢ **To optimize core allocation for a profile:**

1. Open the SBC General Settings page (**Setup** menu > **Signaling & Media** tab > **SBC** folder > **SBC General Settings**).
2. From the 'SBC Performance Profile' drop-down list, select the required profile:


3. Click **Apply**, and then reset the device with a burn-to-flash for your settings to take effect.

**This page is intentionally left blank.**

# A AudioCodes INI File

The *ini* configuration file of the SBC, corresponding to the Web-based configuration as described in Section 4 on page 17, is shown below:

**Note:** To load or save an *ini* file, use the Configuration File page (**Setup** menu > **Administration** tab > **Maintenance** folder > **Configuration File**).

```
;**************
;** Ini File **
;**************

;Board: Mediant VE-H SBC
;HW Board Type: 73 FK Board Type: 80
;Serial Number: 126153906829976
;Slot Number: 1
;Software Version: 7.20A.204.128
;ISO Version: Mediant Software E-SBC (ver 7.20A.204.015)
;DSP Software Version: SOFTDSP => 710.08
;Board IP Address: 172.31.13.46
;Board Subnet Mask: 255.255.240.0
;Board Default Gateway: 172.31.0.1
;Ram size: 7351M   Flash size: 0M
;Num of DSP Cores: 1  Num DSP Channels: 1022
;Profile: NONE
;Client defaults file is being used (file length=387)
;;;Key features:;Board Type: Mediant VE-H SBC ;Coders: G723 G729 G728
NETCODER GSM-FR GSM-EFR AMR EVRC-QCELP G727 ILBC EVRC-B AMR-WB G722 EG711
MS_RTA_NB MS_RTA_WB SILK_NB SILK_WB SPEEX_NB SPEEX_WB OPUS_NB OPUS_WB
;Channel Type: RTP DspCh=100 IPMediaDspCh=100 ;HA ;Security: IPSEC
MediaEncryption StrongEncryption EncryptControlProtocol ;DSP Voice
features: RTCP-XR ;IP Media: VXML ;QOE features: VoiceQualityMonitoring
MediaEnhancement ;Control Protocols: HttpProxy MGCP SIP SBC=100 MSFT CLI
FEU=100 TestCall=100 EMS ;Default features:;Coders: G711 G726;


;MAC Addresses in use:
;---------------------------
;GROUP_1 - 0e:83:f4:76:98:92
;-----------------------------------------------


[SYSTEM Params]

SyslogServerIP = 10.10.10.10
EnableSyslog = 1
HALocalMAC = '0e83f4769892'
TR069ACSPASSWORD = '$1$gQ=='
TR069CONNECTIONREQUESTPASSWORD = '$1$gQ=='
NTPServerIP = 'pool.ntp.org'
;LastConfigChangeTime is hidden but has non-default value
;TLSPkeyPassphrases is hidden but has non-default value
;LocalTimeZoneName is hidden but has non-default value
```

```
[BSP Params]

PCMLawSelect = 3
ARPTableMaxEntries = 3408
UdpPortSpacing = 5
EnterCpuOverloadPercent = 99
ExitCpuOverloadPercent = 95
SbcPerformanceProfile = 2

[ControlProtocols Params]

AdminStateLockControl = 0

[Voice Engine Params]

ENABLEMEDIASECURITY = 1
SbcClusterMode = 0
SbcDeviceRole = 0
PLThresholdLevelsPerMille_0 = 5
PLThresholdLevelsPerMille_1 = 10
PLThresholdLevelsPerMille_2 = 20
PLThresholdLevelsPerMille_3 = 50

[WEB Params]

Languages = 'en-US', '', '', '', '', '', '', '', '', ''

[SIP Params]

GWDEBUGLEVEL = 5
MSLDAPPRIMARYKEY = 'telephoneNumber'
SBCPREFERENCESMODE = 1
SBCFORKINGHANDLINGMODE = 1
ENERGYDETECTORCMD = 104
ANSWERDETECTORCMD = 12582952
;GWAPPCONFIGURATIONVERSION is hidden but has non-default value

[SNMP Params]


[ PhysicalPortsTable ]

FORMAT PhysicalPortsTable_Index = PhysicalPortsTable_Port,
PhysicalPortsTable_Mode, PhysicalPortsTable_SpeedDuplex,
PhysicalPortsTable_PortDescription, PhysicalPortsTable_GroupMember,
PhysicalPortsTable_GroupStatus;
PhysicalPortsTable 0 = "GE_1", 1, 4, "User Port #0", "GROUP_1", "Active";

[ \PhysicalPortsTable ]


[ EtherGroupTable ]

FORMAT EtherGroupTable_Index = EtherGroupTable_Group,
EtherGroupTable_Mode, EtherGroupTable_Member1, EtherGroupTable_Member2;
EtherGroupTable 0 = "GROUP_1", 1, "GE_1", "";
EtherGroupTable 1 = "GROUP_2", 0, "", "";
```

```
EtherGroupTable 2 = "GROUP_3", 0, "", "";
EtherGroupTable 3 = "GROUP_4", 0, "", "";
EtherGroupTable 4 = "GROUP_5", 0, "", "";
EtherGroupTable 5 = "GROUP_6", 0, "", "";
EtherGroupTable 6 = "GROUP_7", 0, "", "";
EtherGroupTable 7 = "GROUP_8", 0, "", "";
EtherGroupTable 8 = "GROUP_9", 0, "", "";
EtherGroupTable 9 = "GROUP_10", 0, "", "";
EtherGroupTable 10 = "GROUP_11", 0, "", "";
EtherGroupTable 11 = "GROUP_12", 0, "", "";
EtherGroupTable 12 = "GROUP_13", 0, "", "";
EtherGroupTable 13 = "GROUP_14", 0, "", "";
EtherGroupTable 14 = "GROUP_15", 0, "", "";

[ \EtherGroupTable ]


[ DeviceTable ]

FORMAT DeviceTable_Index = DeviceTable_VlanID,
DeviceTable_UnderlyingInterface, DeviceTable_DeviceName,
DeviceTable_Tagging, DeviceTable_MTU;
DeviceTable 0 = 1, "GROUP_1", "vlan 1", 0, 1500;

[ \DeviceTable ]


[ InterfaceTable ]

FORMAT InterfaceTable_Index = InterfaceTable_ApplicationTypes,
InterfaceTable_InterfaceMode, InterfaceTable_IPAddress,
InterfaceTable_PrefixLength, InterfaceTable_Gateway,
InterfaceTable_InterfaceName, InterfaceTable_PrimaryDNSServerIPAddress,
InterfaceTable_SecondaryDNSServerIPAddress,
InterfaceTable_UnderlyingDevice;
InterfaceTable 0 = 6, 10, 172.31.13.46, 20, 172.31.0.1, "eth0",
172.31.0.2, 0.0.0.0, "vlan 1";

[ \InterfaceTable ]


[ WebUsers ]

FORMAT WebUsers_Index = WebUsers_Username, WebUsers_Password,
WebUsers_Status, WebUsers_PwAgeInterval, WebUsers_SessionLimit,
WebUsers_CliSessionLimit, WebUsers_SessionTimeout, WebUsers_BlockTime,
WebUsers_UserLevel, WebUsers_PwNonce, WebUsers_SSHPublicKey;
WebUsers 0 = "Admin",
"$1$LhkXUQEKVlADUwVcDwMLBAwPDXQlciEnICRyfXt/LXh7L341ZmQwbDNgbzk/b285aGc9A
gdUVFEBX19bUQxTVQk=", 1, 0, 5, -1, 15, 60, 200,
"d1a55c4272f5c6dec325950b740ee574", "";
WebUsers 1 = "User",
"$1$BzE4bT9vaWw5PnZ1ICAncS92Kyh7IiotfCxBQEEQQkEeER0bGBlIGh0cAVRTVwQAA1FaD
1hfDlhcW3IjdScidCI=", 1, 0, 5, -1, 15, 60, 50,
"530b45709d5162a9b999ab8d164c6920", "";

[ \WebUsers ]
```

```
[ TLSContexts ]

FORMAT TLSContexts_Index = TLSContexts_Name, TLSContexts_TLSVersion,
TLSContexts_DTLSVersion, TLSContexts_ServerCipherString,
TLSContexts_ClientCipherString, TLSContexts_RequireStrictCert,
TLSContexts_OcspEnable, TLSContexts_OcspServerPrimary,
TLSContexts_OcspServerSecondary, TLSContexts_OcspServerPort,
TLSContexts_OcspDefaultResponse, TLSContexts_DHKeySize;
TLSContexts 0 = "default", 0, 0, "RC4:AES128", "DEFAULT", 0, 0, , , 2560,
0, 1024;
TLSContexts 1 = "Teams", 4, 0, "RC4:AES128", "DEFAULT", 0, 0, 0.0.0.0,
0.0.0.0, 2560, 0, 1024;

[ \TLSContexts ]


[ AudioCodersGroups ]

FORMAT AudioCodersGroups_Index = AudioCodersGroups_Name;
AudioCodersGroups 0 = "AudioCodersGroups_0";
AudioCodersGroups 1 = "AudioCodersGroups_1";

[ \AudioCodersGroups ]


[ IpProfile ]

FORMAT IpProfile_Index = IpProfile_ProfileName, IpProfile_IpPreference,
IpProfile_CodersGroupName, IpProfile_IsFaxUsed,
IpProfile_JitterBufMinDelay, IpProfile_JitterBufOptFactor,
IpProfile_IPDiffServ, IpProfile_SigIPDiffServ,
IpProfile_RTPRedundancyDepth, IpProfile_CNGmode,
IpProfile_VxxTransportType, IpProfile_NSEMode, IpProfile_IsDTMFUsed,
IpProfile_PlayRBTone2IP, IpProfile_EnableEarlyMedia,
IpProfile_ProgressIndicator2IP, IpProfile_EnableEchoCanceller,
IpProfile_CopyDest2RedirectNumber, IpProfile_MediaSecurityBehaviour,
IpProfile_CallLimit, IpProfile_DisconnectOnBrokenConnection,
IpProfile_FirstTxDtmfOption, IpProfile_SecondTxDtmfOption,
IpProfile_RxDTMFOption, IpProfile_EnableHold, IpProfile_InputGain,
IpProfile_VoiceVolume, IpProfile_AddIEInSetup,
IpProfile_SBCExtensionCodersGroupName,
IpProfile_MediaIPVersionPreference, IpProfile_TranscodingMode,
IpProfile_SBCAllowedMediaTypes, IpProfile_SBCAllowedAudioCodersGroupName,
IpProfile_SBCAllowedVideoCodersGroupName, IpProfile_SBCAllowedCodersMode,
IpProfile_SBCMediaSecurityBehaviour, IpProfile_SBCRFC2833Behavior,
IpProfile_SBCAlternativeDTMFMethod, IpProfile_SBCSendMultipleDTMFMethods,
IpProfile_SBCAssertIdentity, IpProfile_AMDSensitivityParameterSuit,
IpProfile_AMDSensitivityLevel, IpProfile_AMDMaxGreetingTime,
IpProfile_AMDMaxPostSilenceGreetingTime, IpProfile_SBCDiversionMode,
IpProfile_SBCHistoryInfoMode, IpProfile_EnableQSIGTunneling,
IpProfile_SBCFaxCodersGroupName, IpProfile_SBCFaxBehavior,
IpProfile_SBCFaxOfferMode, IpProfile_SBCFaxAnswerMode,
IpProfile_SbcPrackMode, IpProfile_SBCSessionExpiresMode,
IpProfile_SBCRemoteUpdateSupport, IpProfile_SBCRemoteReinviteSupport,
IpProfile_SBCRemoteDelayedOfferSupport, IpProfile_SBCRemoteReferBehavior,
IpProfile_SBCRemote3xxBehavior, IpProfile_SBCRemoteMultiple18xSupport,
IpProfile_SBCRemoteEarlyMediaResponseType,
IpProfile_SBCRemoteEarlyMediaSupport, IpProfile_EnableSymmetricMKI,
IpProfile_MKISize, IpProfile_SBCEnforceMKISize,
IpProfile_SBCRemoteEarlyMediaRTP, IpProfile_SBCRemoteSupportsRFC3960,
IpProfile_SBCRemoteCanPlayRingback, IpProfile_EnableEarly183,
IpProfile_EarlyAnswerTimeout, IpProfile_SBC2833DTMFPayloadType,
IpProfile_SBCUserRegistrationTime, IpProfile_ResetSRTPStateUponRekey,
```

```
IpProfile_AmdMode, IpProfile_SBCReliableHeldToneSource,
IpProfile_GenerateSRTPKeys, IpProfile_SBCPlayHeldTone,
IpProfile_SBCRemoteHoldFormat, IpProfile_SBCRemoteReplacesBehavior,
IpProfile_SBCSDPPtimeAnswer, IpProfile_SBCPreferredPTime,
IpProfile_SBCUseSilenceSupp, IpProfile_SBCRTPRedundancyBehavior,
IpProfile_SBCPlayRBTToTransferee, IpProfile_SBCRTCPMode,
IpProfile_SBCJitterCompensation,
IpProfile_SBCRemoteRenegotiateOnFaxDetection,
IpProfile_JitterBufMaxDelay,
IpProfile_SBCUserBehindUdpNATRegistrationTime,
IpProfile_SBCUserBehindTcpNATRegistrationTime,
IpProfile_SBCSDPHandleRTCPAttribute,
IpProfile_SBCRemoveCryptoLifetimeInSDP, IpProfile_SBCIceMode,
IpProfile_SBCRTCPMux, IpProfile_SBCMediaSecurityMethod,
IpProfile_SBCHandleXDetect, IpProfile_SBCRTCPFeedback,
IpProfile_SBCRemoteRepresentationMode, IpProfile_SBCKeepVIAHeaders,
IpProfile_SBCKeepRoutingHeaders, IpProfile_SBCKeepUserAgentHeader,
IpProfile_SBCRemoteMultipleEarlyDialogs,
IpProfile_SBCRemoteMultipleAnswersMode, IpProfile_SBCDirectMediaTag,
IpProfile_SBCAdaptRFC2833BWToVoiceCoderBW,
IpProfile_CreatedByRoutingServer, IpProfile_SBCFaxReroutingMode,
IpProfile_SBCMaxCallDuration, IpProfile_SBCGenerateRTP,
IpProfile_SBCISUPBodyHandling, IpProfile_SBCISUPVariant,
IpProfile_SBCVoiceQualityEnhancement, IpProfile_SBCMaxOpusBW,
IpProfile_SBCEnhancedPlc, IpProfile_LocalRingbackTone,
IpProfile_LocalHeldTone, IpProfile_SBCGenerateNoOp,
IpProfile_SBCRemoveUnKnownCrypto;
IpProfile 1 = "Colt", 1, "AudioCodersGroups_0", 0, 10, 10, 46, 24, 0, 0,
2, 0, 0, 0, 0, -1, 1, 0, 0, -1, 1, 4, -1, 1, 1, 0, 0, "", "", 0, 0, "",
"", "", 0, 2, 0, 0, 0, 1, 0, 8, 300, 400, 0, 0, 0, "", 0, 0, 1, 3, 0, 2,
2, 1, 3, 2, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1,
0, 0, 0, 0, 1, 0, 0, 0, 300, -1, -1, 0, 0, 0, 0, 0, 0, 0, -1, -1, -1, -1,
-1, 0, "", 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, -1, 0, 0;
IpProfile 2 = "Teams", 1, "AudioCodersGroups_0", 0, 10, 10, 46, 24, 0, 0,
2, 0, 0, 0, 0, -1, 1, 0, 0, -1, 1, 4, -1, 1, 1, 0, 0, "",
"AudioCodersGroups_1", 0, 0, "", "", "", 0, 1, 0, 0, 0, 0, 0, 8, 300,
400, 0, 0, 0, "", 0, 0, 1, 3, 0, 0, 0, 0, 3, 2, 1, 0, 1, 0, 0, 0, 1, 0,
1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 3, 0, 0, 0, 0, 0, 0, 1, 0, 0, 300, -1, -1,
0, 0, 1, 0, 0, 0, 0, -1, -1, -1, -1, -1, 0, "", 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, -1, -1, 0, 0;

[ \IpProfile ]


[ CpMediaRealm ]

FORMAT CpMediaRealm_Index = CpMediaRealm_MediaRealmName,
CpMediaRealm_IPv4IF, CpMediaRealm_IPv6IF, CpMediaRealm_RemoteIPv4IF,
CpMediaRealm_RemoteIPv6IF, CpMediaRealm_PortRangeStart,
CpMediaRealm_MediaSessionLeg, CpMediaRealm_PortRangeEnd,
CpMediaRealm_IsDefault, CpMediaRealm_QoeProfile, CpMediaRealm_BWProfile,
CpMediaRealm_TopologyLocation;
CpMediaRealm 0 = "Colt", "eth0", "", "", "", 6000, 100, 6499, 0, "", "",
1;
CpMediaRealm 1 = "Teams", "eth0", "", "", "", 7000, 100, 7499, 0, "", "",
0;

[ \CpMediaRealm ]


[ SBCRoutingPolicy ]
```

```
FORMAT SBCRoutingPolicy_Index = SBCRoutingPolicy_Name,
SBCRoutingPolicy_LCREnable, SBCRoutingPolicy_LCRAverageCallLength,
SBCRoutingPolicy_LCRDefaultCost, SBCRoutingPolicy_LdapServerGroupName;
SBCRoutingPolicy 0 = "Default_SBCRoutingPolicy", 0, 1, 0, "";

[ \SBCRoutingPolicy ]


[ SRD ]

FORMAT SRD_Index = SRD_Name, SRD_BlockUnRegUsers, SRD_MaxNumOfRegUsers,
SRD_EnableUnAuthenticatedRegistrations, SRD_SharingPolicy,
SRD_UsedByRoutingServer, SRD_SBCOperationMode, SRD_SBCRoutingPolicyName,
SRD_SBCDialPlanName, SRD_AdmissionProfile;
SRD 0 = "DefaultSRD", 0, -1, 1, 0, 0, 0, "Default_SBCRoutingPolicy", "",
"";

[ \SRD ]


[ MessagePolicy ]

FORMAT MessagePolicy_Index = MessagePolicy_Name,
MessagePolicy_MaxMessageLength, MessagePolicy_MaxHeaderLength,
MessagePolicy_MaxBodyLength, MessagePolicy_MaxNumHeaders,
MessagePolicy_MaxNumBodies, MessagePolicy_SendRejection,
MessagePolicy_MethodList, MessagePolicy_MethodListType,
MessagePolicy_BodyList, MessagePolicy_BodyListType,
MessagePolicy_UseMaliciousSignatureDB;
MessagePolicy 0 = "Malicious Signature DB Protection", -1, -1, -1, -1, -
1, 1, "", 0, "", 0, 1;

[ \MessagePolicy ]


[ SIPInterface ]

FORMAT SIPInterface_Index = SIPInterface_InterfaceName,
SIPInterface_NetworkInterface, SIPInterface_ApplicationType,
SIPInterface_UDPPort, SIPInterface_TCPPort, SIPInterface_TLSPort,
SIPInterface_AdditionalUDPPorts, SIPInterface_AdditionalUDPPortsMode,
SIPInterface_SRDName, SIPInterface_MessagePolicyName,
SIPInterface_TLSContext, SIPInterface_TLSMutualAuthentication,
SIPInterface_TCPKeepaliveEnable,
SIPInterface_ClassificationFailureResponseType,
SIPInterface_PreClassificationManSet, SIPInterface_EncapsulatingProtocol,
SIPInterface_MediaRealm, SIPInterface_SBCDirectMedia,
SIPInterface_BlockUnRegUsers, SIPInterface_MaxNumOfRegUsers,
SIPInterface_EnableUnAuthenticatedRegistrations,
SIPInterface_UsedByRoutingServer, SIPInterface_TopologyLocation,
SIPInterface_PreParsingManSetName, SIPInterface_AdmissionProfile,
SIPInterface_CallSetupRulesSetId;
SIPInterface 0 = "Colt", "eth0", 2, 5060, 0, 0, "", 0, "DefaultSRD",
"Malicious Signature DB Protection", "default", -1, 0, 0, -1, 0, "Colt",
0, -1, -1, -1, 0, 1, "", "", -1;
SIPInterface 1 = "Teams", "eth0", 2, 0, 0, 5061, "", 0, "DefaultSRD",
"Malicious Signature DB Protection", "Teams", -1, 1, 0, -1, 0, "Teams",
0, -1, -1, -1, 0, 0, "", "", -1;

[ \SIPInterface ]
```

```
[ ProxySet ]

FORMAT ProxySet_Index = ProxySet_ProxyName,
ProxySet_EnableProxyKeepAlive, ProxySet_ProxyKeepAliveTime,
ProxySet_ProxyLoadBalancingMethod, ProxySet_IsProxyHotSwap,
ProxySet_SRDName, ProxySet_ClassificationInput, ProxySet_TLSContextName,
ProxySet_ProxyRedundancyMode, ProxySet_DNSResolveMethod,
ProxySet_KeepAliveFailureResp, ProxySet_GWIPv4SIPInterfaceName,
ProxySet_SBCIPv4SIPInterfaceName, ProxySet_GWIPv6SIPInterfaceName,
ProxySet_SBCIPv6SIPInterfaceName, ProxySet_MinActiveServersLB,
ProxySet_SuccessDetectionRetries, ProxySet_SuccessDetectionInterval,
ProxySet_FailureDetectionRetransmissions;
ProxySet 0 = "ProxySet_0", 0, 60, 0, 0, "DefaultSRD", 0, "", -1, -1, "",
"", "Colt", "", "", 1, 1, 10, -1;
ProxySet 1 = "Colt", 1, 60, 0, 0, "DefaultSRD", 0, "", -1, -1, "", "",
"Colt", "", "", 1, 1, 10, -1;
ProxySet 2 = "Teams", 1, 60, 2, 1, "DefaultSRD", 0, "Teams", -1, 1, "",
"", "Teams", "", "", 1, 1, 10, -1;

[ \ProxySet ]


[ IPGroup ]

FORMAT IPGroup_Index = IPGroup_Type, IPGroup_Name, IPGroup_ProxySetName,
IPGroup_SIPGroupName, IPGroup_ContactUser, IPGroup_SipReRoutingMode,
IPGroup_AlwaysUseRouteTable, IPGroup_SRDName, IPGroup_MediaRealm,
IPGroup_ClassifyByProxySet, IPGroup_ProfileName,
IPGroup_MaxNumOfRegUsers, IPGroup_InboundManSet, IPGroup_OutboundManSet,
IPGroup_RegistrationMode, IPGroup_AuthenticationMode, IPGroup_MethodList,
IPGroup_EnableSBCClientForking, IPGroup_SourceUriInput,
IPGroup_DestUriInput, IPGroup_ContactName, IPGroup_Username,
IPGroup_Password, IPGroup_UUIFormat, IPGroup_QOEProfile,
IPGroup_BWProfile, IPGroup_AlwaysUseSourceAddr, IPGroup_MsgManUserDef1,
IPGroup_MsgManUserDef2, IPGroup_SIPConnect, IPGroup_SBCPSAPMode,
IPGroup_DTLSContext, IPGroup_CreatedByRoutingServer,
IPGroup_UsedByRoutingServer, IPGroup_SBCOperationMode,
IPGroup_SBCRouteUsingRequestURIPort, IPGroup_SBCKeepOriginalCallID,
IPGroup_TopologyLocation, IPGroup_SBCDialPlanName,
IPGroup_CallSetupRulesSetId, IPGroup_Tags, IPGroup_SBCUserStickiness,
IPGroup_UserUDPPortAssignment, IPGroup_AdmissionProfile,
IPGroup_ProxyKeepAliveUsingIPG;
IPGroup 0 = 0, "Default_IPG", "ProxySet_0", "", "", -1, 0, "DefaultSRD",
"", 1, "", -1, -1, -1, 0, 0, "", 0, -1, -1, "", "", "$1$gQ==", 0, "", "",
0, "", "", 0, 0, "default", 0, 0, -1, 0, 0, 1, "", -1, "", 0, 0, "", 0;
IPGroup 1 = 0, "Colt", "Colt", "", "", -1, 0, "DefaultSRD", "Colt", 1,
"Colt", -1, -1, 4, 0, 0, "", 0, -1, -1, "", "", "$1$gQ==", 0, "", "", 0,
"", "", 0, 0, "default", 0, 0, -1, 0, 0, 1, "", -1, "", 0, 0, "", 0;
IPGroup 2 = 0, "Teams", "Teams", "int-sbc2.audctrunk.aceducation.info",
"", -1, 0, "DefaultSRD", "Teams", 0, "Teams", -1, -1, -1, 0, 0, "", 0, -
1, -1, "int-sbc2.audctrunk.aceducation.info", "", "$1$gQ==", 0, "", "",
1, "", "", 0, 0, "Teams", 0, 0, -1, 0, 0, 0, "", -1, "", 0, 0, "", 1;

[ \IPGroup ]


[ Srv2Ip ]

FORMAT Srv2Ip_Index = Srv2Ip_InternalDomain, Srv2Ip_TransportType,
Srv2Ip_Dns1, Srv2Ip_Priority1, Srv2Ip_Weight1, Srv2Ip_Port1, Srv2Ip_Dns2,
Srv2Ip_Priority2, Srv2Ip_Weight2, Srv2Ip_Port2, Srv2Ip_Dns3,
Srv2Ip_Priority3, Srv2Ip_Weight3, Srv2Ip_Port3;
```

```
Srv2Ip 0 = "teams.local", 2, "sip.pstnhub.microsoft.com", 1, 1, 5061,
"sip2.pstnhub.microsoft.com", 2, 1, 5061, "sip3.pstnhub.microsoft.com",
3, 1, 5061;

[ \Srv2Ip ]


[ ProxyIp ]

FORMAT ProxyIp_Index = ProxyIp_ProxySetId, ProxyIp_ProxyIpIndex,
ProxyIp_IpAddress, ProxyIp_TransportType, ProxyIp_Priority,
ProxyIp_Weight;
ProxyIp 0 = "1", 0, "80.169.151.6:5060", 0, 0, 0;
ProxyIp 1 = "2", 0, "teams.local", 2, 0, 0;

[ \ProxyIp ]


[ ConditionTable ]

FORMAT ConditionTable_Index = ConditionTable_Name,
ConditionTable_Condition;
ConditionTable 0 = "Teams-Contact", "Header.Contact.URL.Host contains
'pstnhub.microsoft.com'";

[ \ConditionTable ]


[ IP2IPRouting ]

FORMAT IP2IPRouting_Index = IP2IPRouting_RouteName,
IP2IPRouting_RoutingPolicyName, IP2IPRouting_SrcIPGroupName,
IP2IPRouting_SrcUsernamePrefix, IP2IPRouting_SrcHost,
IP2IPRouting_DestUsernamePrefix, IP2IPRouting_DestHost,
IP2IPRouting_RequestType, IP2IPRouting_MessageConditionName,
IP2IPRouting_ReRouteIPGroupName, IP2IPRouting_Trigger,
IP2IPRouting_CallSetupRulesSetId, IP2IPRouting_DestType,
IP2IPRouting_DestIPGroupName, IP2IPRouting_DestSIPInterfaceName,
IP2IPRouting_DestAddress, IP2IPRouting_DestPort,
IP2IPRouting_DestTransportType, IP2IPRouting_AltRouteOptions,
IP2IPRouting_GroupPolicy, IP2IPRouting_CostGroup, IP2IPRouting_DestTags,
IP2IPRouting_SrcTags, IP2IPRouting_IPGroupSetName,
IP2IPRouting_RoutingTagName, IP2IPRouting_InternalAction;
IP2IPRouting 0 = "Terminate OPTIONS", "Default_SBCRoutingPolicy", "Any",
"*", "*", "*", "*", 6, "", "Any", 0, -1, 1, "", "", "internal", 0, -1, 0,
0, "", "", "", "", "default", "";
IP2IPRouting 1 = "Refer from Teams", "Default_SBCRoutingPolicy", "Any",
"*", "*", "*", "*", 0, "", "Teams", 2, -1, 2, "Teams", "", "", 0, -1, 0,
0, "", "", "", "", "default", "";
IP2IPRouting 2 = "Teams to Colt", "Default_SBCRoutingPolicy", "Teams",
"*", "*", "*", "*", 0, "", "Any", 0, -1, 0, "Colt", "", "", 0, -1, 0, 0,
"", "", "", "", "default", "";
IP2IPRouting 3 = "Colt to Teams", "Default_SBCRoutingPolicy", "Colt",
"*", "*", "*", "*", 0, "", "Any", 0, -1, 0, "Teams", "", "", 0, -1, 0, 0,
"", "", "", "", "default", "";

[ \IP2IPRouting ]


[ Classification ]
```

```
FORMAT Classification_Index = Classification_ClassificationName,
Classification_MessageConditionName, Classification_SRDName,
Classification_SrcSIPInterfaceName, Classification_SrcAddress,
Classification_SrcPort, Classification_SrcTransportType,
Classification_SrcUsernamePrefix, Classification_SrcHost,
Classification_DestUsernamePrefix, Classification_DestHost,
Classification_ActionType, Classification_SrcIPGroupName,
Classification_DestRoutingPolicy, Classification_IpProfileName,
Classification_IPGroupSelection, Classification_IpGroupTagName;
Classification 0 = "Teams", "Teams-Contact", "DefaultSRD", "Teams", "",
0, -1, "*", "*", "*", "int-sbc2.audctrunk.aceducation.info", 1, "Teams",
"", "", 0, "default";

[ \Classification ]


[ MessageManipulations ]

FORMAT MessageManipulations_Index =
MessageManipulations_ManipulationName, MessageManipulations_ManSetID,
MessageManipulations_MessageType, MessageManipulations_Condition,
MessageManipulations_ActionSubject, MessageManipulations_ActionType,
MessageManipulations_ActionValue, MessageManipulations_RowRole;
MessageManipulations 0 = "Call Forward", 4, "", "Header.History-Info
exists", "Header.Diversion", 0, "Header.History-Info.HistoryInfo", 0;
MessageManipulations 1 = "Call Forward", 4, "", "", "Header.History-
Info", 1, "", 1;
MessageManipulations 2 = "Change Reason Cause Code to 21", 4,
"Any.Response", "Header.Reason exists", "Header.Reason.Reason.Cause", 2,
"'21'", 0;

[ \MessageManipulations ]


[ NATTranslation ]

FORMAT NATTranslation_Index = NATTranslation_SrcIPInterfaceName,
NATTranslation_RemoteInterfaceName, NATTranslation_TargetIpMode,
NATTranslation_TargetIPAddress, NATTranslation_SourceStartPort,
NATTranslation_SourceEndPort, NATTranslation_TargetStartPort,
NATTranslation_TargetEndPort;
NATTranslation 0 = "eth0", "", 1, "", "5060", "5060", "5060", "5060";
NATTranslation 1 = "eth0", "", 1, "", "6000", "6500", "6000", "6500";
NATTranslation 2 = "eth0", "", 1, "", "5061", "5061", "5061", "5061";
NATTranslation 3 = "eth0", "", 1, "", "7000", "7500", "7000", "7500";

[ \NATTranslation ]


[ GwRoutingPolicy ]

FORMAT GwRoutingPolicy_Index = GwRoutingPolicy_Name,
GwRoutingPolicy_LCREnable, GwRoutingPolicy_LCRAverageCallLength,
GwRoutingPolicy_LCRDefaultCost, GwRoutingPolicy_LdapServerGroupName;
GwRoutingPolicy 0 = "GwRoutingPolicy", 0, 1, 0, "";

[ \GwRoutingPolicy ]
```

```
[ MaliciousSignatureDB ]

FORMAT MaliciousSignatureDB_Index = MaliciousSignatureDB_Name,
MaliciousSignatureDB_Pattern;
MaliciousSignatureDB 0 = "SIPVicious", "Header.User-Agent.content prefix
'friendly-scanner'";
MaliciousSignatureDB 1 = "SIPScan", "Header.User-Agent.content prefix
'sip-scan'";
MaliciousSignatureDB 2 = "Smap", "Header.User-Agent.content prefix
'smap'";
MaliciousSignatureDB 3 = "Sipsak", "Header.User-Agent.content prefix
'sipsak'";
MaliciousSignatureDB 4 = "Sipcli", "Header.User-Agent.content prefix
'sipcli'";
MaliciousSignatureDB 5 = "Sivus", "Header.User-Agent.content prefix
'SIVuS'";
MaliciousSignatureDB 6 = "Gulp", "Header.User-Agent.content prefix
'Gulp'";
MaliciousSignatureDB 7 = "Sipv", "Header.User-Agent.content prefix
'sipv'";
MaliciousSignatureDB 8 = "Sundayddr Worm", "Header.User-Agent.content
prefix 'sundayddr'";
MaliciousSignatureDB 9 = "VaxIPUserAgent", "Header.User-Agent.content
prefix 'VaxIPUserAgent'";
MaliciousSignatureDB 10 = "VaxSIPUserAgent", "Header.User-Agent.content
prefix 'VaxSIPUserAgent'";
MaliciousSignatureDB 11 = "SipArmyKnife", "Header.User-Agent.content
prefix 'siparmyknife'";
MaliciousSignatureDB 12 = "pplsip", "Header.User-Agent contains
'pplsip'";
MaliciousSignatureDB 13 = "zxcvfdf11", "Header.User-Agent contains
'zxcvfdf11'";

[ \MaliciousSignatureDB ]


[ AudioCoders ]

FORMAT AudioCoders_Index = AudioCoders_AudioCodersGroupId,
AudioCoders_AudioCodersIndex, AudioCoders_Name, AudioCoders_pTime,
AudioCoders_rate, AudioCoders_PayloadType, AudioCoders_Sce,
AudioCoders_CoderSpecific;
AudioCoders 0 = "AudioCodersGroups_0", 0, 1, 2, 90, -1, 0, "";
AudioCoders 1 = "AudioCodersGroups_1", 0, 35, 2, 19, 76, 0, "";
AudioCoders 2 = "AudioCodersGroups_1", 1, 36, 2, 43, 77, 0, "";
AudioCoders 3 = "AudioCodersGroups_1", 2, 1, 2, 90, -1, 0, "";
AudioCoders 4 = "AudioCodersGroups_1", 3, 2, 2, 90, -1, 0, "";
AudioCoders 5 = "AudioCodersGroups_1", 4, 3, 2, 19, -1, 0, "";

[ \AudioCoders ]
```

**This page is intentionally left blank.**

**International Headquarters**
1 Hayarden Street,
Airport City
Lod 7019900, Israel
Tel: +972-3-976-4000
Fax: +972-3-976-4040

**AudioCodes Inc.**
27 World's Fair Drive,
Somerset, NJ 08873
Tel: +1-732-469-0880
Fax: +1-732-469-2298

**Contact us:** https://www.audiocodes.com/corporate/offices-worldwide
**website**: https://www.audiocodes.com



Document #: LTRT-12315