AT&T IP Flexible Reach Services Including MIS/PNT/AVPN Transports with Oracle Enterprise Session Border Controller, Oracle Enterprise Operations Monitor and Microsoft Skype for Business

Technical Application Note

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Table of Contents

Intended Audience

This is a technical document intended for use by Oracle Systems Engineers, third party Systems Integrators, Oracle Enterprise customers and partners and end users of Oracle Enterprise Session Border Controller (E-SBC) as well as service provider based session border controller. It assumes that the reader is familiar with basic operations of Oracle Session Border Controller 3800/4000 and 6000 series platforms.

Document Overview

This Oracle technical application note outlines the recommended configurations for the Oracle Session Border Controller 3800 series for connecting AT&T's IP Flexible Reach service to Microsoft Skype for Business (SFB) customers. The solution contained within this document has been certified on Oracle's Acme Packet OS ECZ 7.3m1p1.

Microsoft Skype for Business offers the ability to connect to SIP based telephony trunks using an IP communications. This reduces the cost and complexity of extending an enterprise telephony system outside its network borders. Oracle Enterprise Session Border Controllers (E-SBCs) play an important role in SIP trunking as they are used by many ITSPs and some enterprises as part of their SIP trunking infrastructure.

This application note has been prepared as a means of ensuring that AT&T's IP Flexible Reach SIP trunking between Microsoft Skype for Business and Oracle E-SBC are configured in the optimal manner.

It should be noted that while this application note focuses on the optimal configurations for the Oracle SBC in an enterprise Skype for Business environment, the same SBC configuration model can also be used for other enterprise SIP trunking applications with changes to the configuration on the ESBC. In addition, it should be noted that the SBC configuration provided in this guide focuses strictly on the Skype for Business associated parameters. Many SBC applications may have additional configuration requirements that are specific to individual customer requirements. These configuration items are not covered in this guide. Please contact your Oracle representative with any questions pertaining to this topic.

Introduction

Audience

This is a technical intended for telecommunications engineers with the purpose of configuring the Oracle Enterprise Session Border Controller (E-SBC) and Skype for Business Server. There will be steps that require navigating Microsoft windows Server as well as the Acme Packet Command Line Interface (ACLI). Understanding the basic concepts of TCP/UDP, IP/Routing and SIP/RTP are also necessary to complete the configuration and for troubleshooting, if necessary.

Requirements

- Fully functioning Skype for Business Server deployment, including Active Directory and DNS
- A dedicated Mediation Server for the SIP trunking connection
- Microsoft Skype for Business 2015 – Version 6.0.93190.0
- Skype for Business 2015 client – Version 15.0.4753.1000
- Oracle Enterprise Session Border Controller AP 3820 or any Oracle ESBC appliance or VM edition running Net-Net OS ECZ730m1p1.32.bz. Note: the configuration running on the SBC is backward/forward compatible with any release in the 7.3.0 stream.

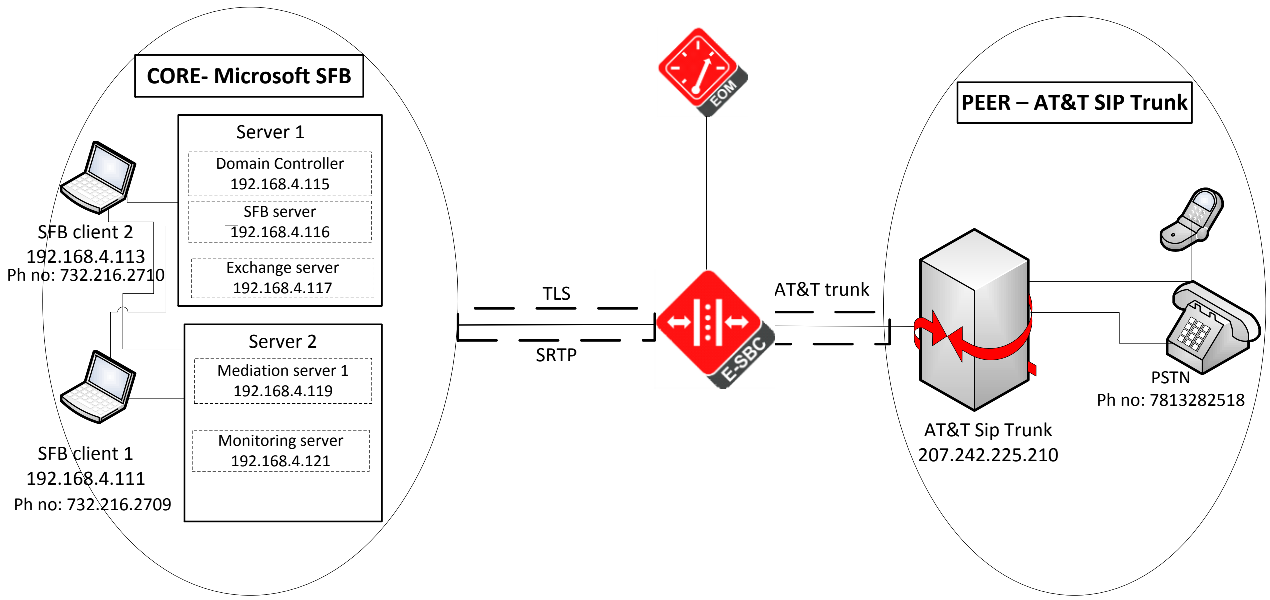
Software Versions Used

The following are the software versions used in this testing.

Component	Version
E-SBC	ECZ7.3.0 MR-1 P1 (Build 134)
Oracle Enterprise Operations Monitor	3.3.90.0.0
Microsoft Skype for Business Server 2015	6.0.9319.0

Lab Configuration

The following diagram illustrates the lab environment created to facilitate certification testing (IP addressing/Port below is only a reference, they can change per your network specifications).



Phase 1 – Configuring the Oracle E-SBC

In this section we describe the steps for configuring a Net-Net E-SBC for use with Skype for Business Server in a SIP trunking scenario.

In Scope

The following Step-by-Step guide configuring the Net-Net E-SBC assumes that this is a newly deployed device dedicated to a single customer.

Note that Oracle Communications offers several products and solutions that can interface with Skype for Business Server. This document covers the setup for the Net-Net E-SBC platforms software SCZ 7.3m1p1 or later. A Net-Net 3800-series (NN3820) platform was used as the platform for developing this guide. If instructions are needed for other Oracle Communications products, please contact your Oracle Communications representative.

Out of Scope

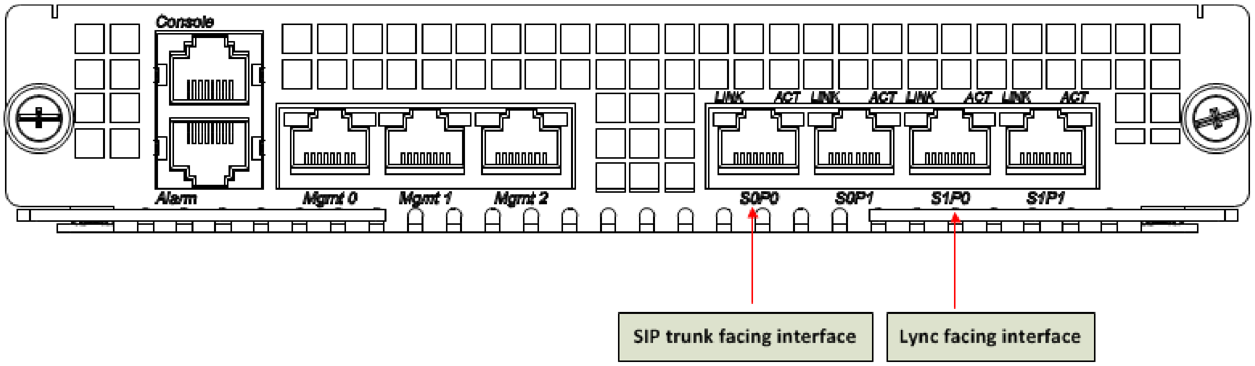
- Configuration of Network management including SNMP and RADIUS

What you will need

- Serial Console cross over cable with RJ-45 connector
- Terminal emulation application such as PuTTYor HyperTerm
- Passwords for the User and Superuser modes on the Net-Net E-SBC
- Signaling IP address and port of Skype for Business Mediation Server
- Signaling and media IP addresses and ports to be used on the Net-Net E-SBC facing Skype for Business and AT&T SIP trunk
- Signaling IP address and port of the next hop network element in the AT&T SIP trunk network
- IP address of the enterprise DNS server

Configuration

Once the Net-Net E-SBC is racked and the power cable connected, you are ready to set up physical network connectivity.



Plug the slot 0 port 0 (s0p0) interface into your SIP trunk provider (SIP trunk facing) network and the slot 1 port 0 (s1p0) interface into your SFB (SFB mediation server-facing) network as shown in the diagram above. Once connected, you are ready to power on and perform the following steps.
All commands are in bold, such as **configure terminal**; parameters in bold red such as **PE11-ATT-Trunk** are parameters which are specific to an individual deployment.

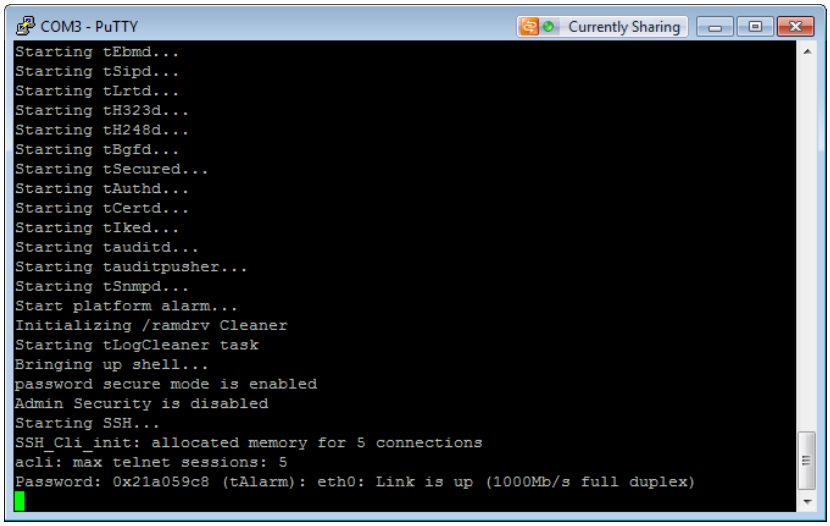
Note: The ACLI is case sensitive.

Establish the serial connection to the Net-Net SBC.

Confirm the Net-Net SD is powered off and connect the serial console cable to the Net-Net SD to a workstation running a terminal emulator application such as PuTTy. Start the terminal emulation application using the following settings:

- Baud Rate=115200
- Data Bits=8
- Parity=None
- Stop Bits=1
- Flow Control=None

Start the Net-Net SD and confirm that you see the following output from the bootup sequence.

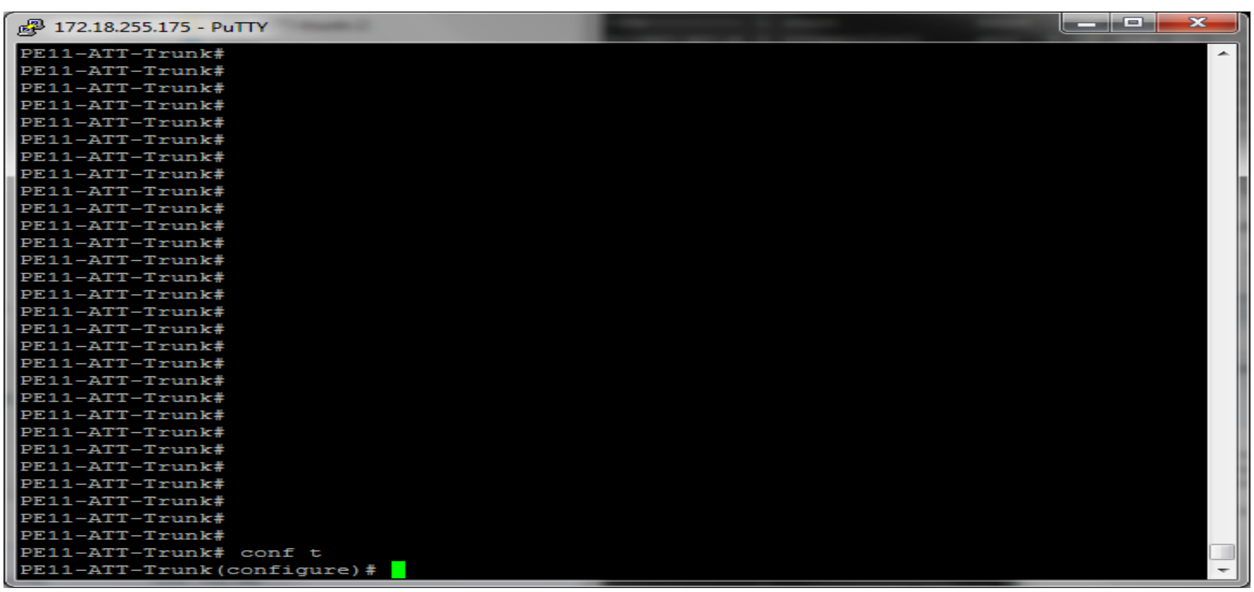


1. **Login to the Net-Net SD and enter the configuration mode**

Enter the following commands to login to the Net-Net SD and move to the configuration mode. Note that the default Net-Net SBC password is "**acme**" and the default super user password is "**packet**".

```
Password: acme
PE11-ATT-Trunk> enable
Password: packet
PE11-ATT-Trunk# configure terminal
PE11-ATT-Trunk (configure)#
```

You are now in the Global Configuration mode.



2. Do the Initial Configuration – Assign the management Interface an IP address

To assign an IP address, one has to configure the bootparams on the Net-Net SD, by going to PE11-ATT-Trunk#configure terminal --- >bootparams

- Once you type "bootparam" you have to use "carriage return" key to navigate down
- A reboot is required if changes are made to the existing bootparams

```
PE11-ATT-Trunk#(configure)bootparam

'.' = clear field;  '-' = go to previous field;  q = quit
boot device             : eth0
processor number        : 0
host name               : acmesystem
file name               : /boot/nnECZ730m1p1.32.bz--- >location where the software is loaded on the SBC
inet on ethernet (e)    : 172.18.255.175:ffff0000 --- > This is the ip address of the management interface of the SBC, type the IP address and mask in hex
inet on backplane (b)   :
host inet (h)           :
gateway inet (g)        : 172.18.0.1 --- > gateway address here
user (u)                : vxftp
ftp password (pw) (blank = use rsh)     : vxftp
flags (f)               :
target name (tn)        : PE11-ATT-Trunk
startup script (s)      :
other (o)               :
```

3. Configure system element values

To configure system element values, use the **system-config** command under the system branch. Then enter values appropriate to your environment, including your default gateway IP address for your management Ethernet interface.

```
PE11-ATT-TRUNK(configure)# system
PE11-ATT-TRUNK(system)# system-config
PE11-ATT-TRUNK(system-config)# hostname ATT-trunk-IOT
PE11-ATT-TRUNK(system-config)# description "SFB with ATT SIP Trunking"
PE11-ATT-TRUNK(system-config)# location "Bedford, MA"
PE11-ATT-TRUNK(system-config)# default-gateway 172.18.0.1
PE11-ATT-Trunk(comm-monitor)# state enabled
PE11-ATT-Trunk(monitor-collector)# address 172.18.255.101
PE11-ATT-TRUNK(system-config)# done
```

Once the **system-config** settings have completed and you enter **done**, the Net-Net SBC will output a complete listing of all current settings. This will apply throughout the rest of the configuration and is a function of the **done** command. Confirm the output reflects the values you just entered as well as any configuration defaults.

```
system-config
        hostname                                ATT-trunk-IOT
        process-log-level                       DEBUG
        comm-monitor
                state                                   enabled
                monitor-collector
                        address
172.18.255.101
        default-gateway                         172.18.0.1
```

4. Configure Physical Interface values

To configure physical Interface values, use the **phy-interface** command under the system branch. To enter the system branch from system-config, you issue the **exit** command then the **phy-interface** command.
You will first configure the slot 0, port 0 interface designated with the name s0p0. This will be the port plugged into your inside (connection to the PSTN gateway) interface.

```
PE11-ATT-TRUNK(system-config)# exit
PE11-ATT-TRUNK(system)# phy-interface
PE11-ATT-TRUNK(phy-interface)# name M00
PE11-ATT-TRUNK(phy-interface)# operation-type media
PE11-ATT-TRUNK(phy-interface)# slot 0
PE11-ATT-TRUNK(phy-interface)# port 0
PE11-ATT-TRUNK(phy-interface)# done
```

Once the **phy-interface** settings have completed for slot 0 port 0 and you enter **done**, the Net-Net SBC will output a complete listing of all current settings. Confirm the output reflects the values you just entered.

```
phy-interface
name                           M00
operation-type                 Media
port                           0
slot                           0
virtual-mac
admin-state                    enabled
auto-negotiation               enabled
duplex-mode                    FULL
speed                          100
overload-protection            disabled
```

You will now configure the slot 1 port 0 phy-interface, specifying the appropriate values. This will be the port plugged into your outside (connection to the mediation server) interface.

```
PE11-ATT-TRUNK(phy-interface)# name M10
PE11-ATT-TRUNK(phy-interface)# operation-type media
PE11-ATT-TRUNK(phy-interface)# slot 1
PE11-ATT-TRUNK(phy-interface)# port 0
PE11-ATT-TRUNK(phy-interface)# done

phy-interface
name                           M10
operation-type                 Media
port                           0
slot                           1
virtual-mac
admin-state                    enabled
auto-negotiation               enabled
duplex-mode                    FULL
speed                          100
overload-protection            disabled
```

6. Configure Network Interface values

To configure Network Interface values, use the **network-interface** command under the system branch. To enter the system branch from **phy-interface**, you issue the **exit** command then the **network-interface** command.

You will first configure the IP characteristics for the M10 interface defined above.

```
PE11-ATT-TRUNK(phy-interface)# exit
PE11-ATT-TRUNK(system)# network-interface
PE11-ATT-TRUNK(network-interface)# name s1p0
PE11-ATT-TRUNK(network-interface)# description "Mediation Server-facing
inside interface"
PE11-ATT-TRUNK(network-interface)# hostname attsbc.partnersfb.com
PE11-ATT-TRUNK(network-interface)# ip-address 192.168.4.130
PE11-ATT-TRUNK(network-interface)# netmask 255.255.255.0
PE11-ATT-TRUNK(network-interface)# gateway 192.168.4.1
PE11-ATT-TRUNK(network-interface)# dns-ip-primary 192.168.4.150
PE11-ATT-TRUNK(network-interface)# dns-domain partnersfb.com
PE11-ATT-TRUNK(network-interface)# add-hip-ip 192.168.4.130
PE11-ATT-TRUNK(network-interface)# add-icmp-ip 192.168.4.130
PE11-ATT-TRUNK(network-interface)# done

network-interface
        name                           s1p0
```

```
        sub-port-id                    0
        description                    Mediation Server-facing inside
interface
        hostname                                attsbc.partnersfb.com
        ip-address                              192.168.4.130
        netmask                                 255.255.255.0
        gateway                                 192.168.4.1
        dns-ip-primary                          192.168.4.150
        dns-domain                              partnersfb.com
        hip-ip-list                             192.168.4.130
        icmp-address                            192.168.4.130
```

You will now configure the slot 0 port 0 subport 0 network-interface, specifying the appropriate values.

```
PE11-ATT-TRUNK(network-interface)# name s0p0
PE11-ATT-TRUNK(network-interface)# description "ATT gateway-facing inside
interface"
PE11-ATT-TRUNK(network-interface)# ip-address 155.212.214.181
PE11-ATT-TRUNK(network-interface)# netmask 255.255.255.0
PE11-ATT-TRUNK(network-interface)# gateway 155.212.214.1
PE11-ATT-TRUNK(network-interface)# add-hip-ip 155.212.214.181
PE11-ATT-TRUNK(network-interface)# add-icmp-ip 155.212.214.181
PE11-ATT-TRUNK(network-interface)# done

network-interface
      name                          s0p0
      sub-port-id                   0
       description                   VoIP gateway-facing inside interface
        name                                    s0p0
        ip-address                              155.212.214.181
        netmask                                 255.255.255.0
        gateway                                 155.212.214.1
        hip-ip-list                             155.212.214.181
        icmp-address                            155.212.214.181
```

5. Configure Global SIP configuration

To configure the Global SIP values, use the **sip-config** command under the **session-router** branch. To enter the session-router branch from **network-interface**, you issue the **exit** command twice, followed by the **sip-config** command.

```
PE11-ATT-TRUNK(network-interface)# exit
PE11-ATT-TRUNK(system)# exit
PE11-ATT-TRUNK(configure)# session-router
PE11-ATT-TRUNK(session-router)# sip-config
PE11-ATT-TRUNK(sip-config)# home-realm-id core
```

```
PE11-ATT-TRUNK(sip-config)# sip-message-len 6000
PE11-ATT-TRUNK(sip-config)#options +max-udp-length=0
PE11-ATT-TRUNK(sip-config)# done

sip-config
        state                               enabled
        home-realm-id                               core
        options                                     max-udp-length=0
        sip-message-len                             6000
        refer-src-routing                           enabled
```

6. Configure Global Media configuration

To configure the Media values, use the media-manager command under the **media-manager** branch. To enter the **media-manager** branch from **sip-config**, you issue the **exit** command twice, followed by the media-manager command twice.

By issuing the select then done commands at this level, you will be creating the **media-manager** element, enabling the media management functions in the Net-Net SBC with the default values.

```
PE11-ATT-TRUNK(sip-config)# exit
PE11-ATT-TRUNK(session-router)# exit
PE11-ATT-TRUNK(configure)# media-manager
PE11-ATT-TRUNK(media-manager)# media-manager
PE11-ATT-TRUNK(media-manager)# select
PE11-ATT-TRUNK(media-manager)# state enabled
PE11-ATT-TRUNK(media-manager-config)# done

media-manager
state                               enabled
```

7. Configure Realms configuration

To configure the realm values, use the **realm-config** command under the **media-manager** branch. To enter the **media-manager** branch from **media-manager-config**, you issue the **exit** command, followed by the **realm-config** command.

You will create two realms:

- The core, which represents the mediation server-facing (inside) network; and
- The trunk-side, which represents the gateway-facing (outside) network.

```
PE11-ATT-TRUNK(media-manager-config)# exit
PE11-ATT-TRUNK(media-manager)# realm-config
PE11-ATT-TRUNK(realm-config)# identifier core
PE11-ATT-TRUNK(realm-config)# description "Mediation Server-facing
(Inside)"
PE11-ATT-TRUNK(realm-config)# network-interfaces s1p0:0
PE11-ATT-TRUNK(realm-config)# mm-in-realm enabled
PE11-ATT-TRUNK(realm-config)# media-sec-policy sdespolicy
```

```
PE11-ATT-TRUNK(realm-config)# restricted-latching sdp
PE11-ATT-TRUNK(realm-config)# refer-call-transfer enabled
PE11-ATT-TRUNK(realm-config)# codec-policy TrunkCodecs
PE11-ATT-TRUNK(realm-config)# done

realm-config
        identifier                              core
        description                             Mediation Server-facing
(Inside)
        network-interfaces                      s1p0:0
        mm-in-realm                             enabled
        media-sec-policy                        sdespolicy
        restricted-latching                     sdp
        refer-call-transfer                     enabled
        codec-policy                            TrunkCodecs
```

You will now configure the PSTN realm for SIP Trunk side of the SBC, specifying the appropriate values.

```
PE11-ATT-TRUNK(realm-config)# identifier trunk-side
PE11-ATT-TRUNK(realm-config)# description "Gateway (outside)"
PE11-ATT-TRUNK(realm-config)# network-interfaces s0p0:0
PE11-ATT-TRUNK(realm-config)# mm-in-realm enabled
PE11-ATT-TRUNK(realm-config)# media-sec-policy rtponly
PE11-ATT-TRUNK(realm-config)# done

realm-config
        identifier                              trunk-side
        description                             Gateway (outside)
        network-interfaces                      s0p0:0
        mm-in-realm                             enabled
        media-sec-policy                        rtponly
```

8. Configure SIP signaling configuration

To configure the SIP signaling values, use the **sip-interface** command under the **session-router** branch. To enter the **session-router** branch from **realm-config**, you issue the **exit** command twice, followed by the **sip-interface** command.
Here you will be configuring the IP addresses and TCP ports on which the Net-Net SBC will listen for and transmit SIP messages. These will be the same IP addresses as configured on the associated **network-interface** elements.

```
PE11-ATT-TRUNK(realm-config)# exit
PE11-ATT-TRUNK(media-manager)# exit
```

```
PE11-ATT-TRUNK(configure)# session-router
PE11-ATT-TRUNK(session-router)# sip-interface
PE11-ATT-TRUNK(sip-interface)# realm trunk-side
PE11-ATT-TRUNK(sip-interface)# sip-ports
PE11-ATT-TRUNK(sip-port)# address 155.212.214.181
PE11-ATT-TRUNK(sip-port)# allow-anonymous agents-only
PE11-ATT-TRUNK(sip-port)# done
PE11-ATT-TRUNK(sip-port)# exit
PE11-ATT-TRUNK(sip-interface)# out-manipulationid ChangeforPAIandNAT
PE11-ATT-TRUNK(sip-interface)# rfc2833-payload 100
PE11-ATT-TRUNK(sip-interface)# response-map change183to180
PE11-ATT-TRUNK(sip-interface)# done

sip-interface
        realm-id                                trunk-side
        sip-port
                address                           155.212.214.181
                allow-anonymous                   agents-only
        out-manipulationid                      ChangeforPAIandNAT
        rfc2833-payload                         100
        response-map                            change183to180
```

You will now configure the mediation server-facing SIP interface.

```
PE11-ATT-TRUNK(sip-interface)# realm-id core
PE11-ATT-TRUNK(sip-interface)# description "Mediation Server-Facing
(Inside)"
PE11-ATT-TRUNK(sip-interface)# sip-ports
PE11-ATT-TRUNK(sip-port)# address 192.168.4.130
PE11-ATT-TRUNK(sip-port)# transport-protocol TLS
PE11-ATT-TRUNK(sip-port)# port 5067
PE11-ATT-TRUNK(sip-port)# allow-anonymous agents-only
PE11-ATT-TRUNK(sip-port)# done
sip-port
                address                                 192.168.4.130
                port                                    5067
                transport-protocol                      TLS
                tls-profile                             core
                allow-anonymous                         agents-only
PE11-ATT-TRUNK(sip-port)# exit
PE11-ATT-TRUNKPE11-ATT-TRUNK(sip-interface)# done

sip-interface
        state                           enabled
```

```
        realm-id                           core
        description                        Mediation Server-Facing(Inside)
        sip-port
                address                            192.168.4.130
                port                               5067
                transport-protocol                 TLS
                tls-profile                        core
                allow-anonymous                    agents-only
```

9. Configure next-hop signaling elements

To configure the next-hop signaling elements (i.e., the mediation server and PSTN gateway) you define **session-agents.** Use the **session-agent** command under the **session-router** branch. To enter the **session-agent** branch from **sip-interface**, you issue the **exit** command, followed by the **session-agent** command. Here you will be configuring the IP addresses and TCP ports to which the Net-Net SBC will send and from which it will expect to receive SIP messages for your next-hop signaling elements.
SFB Gateway specification outlines the need for the SBC to have capability to do DNS load balancing among a pool of mediation servers. This is currently supported by the ESBC via A or SRV records, however not necessarily in a round-robin manner. In this document and testing, the SBC load balances between two mediation servers that are defined in a group (session-group) with round-robin algorithm configured. It is assumed that when using this kind of a configuration at any point another mediation server is added to the pool of servers, it will need to be explicitly configured on the SBC and added to the **session-group** which will be the responsibility of the enterprise network administrator.

We will first configure the PSTN gateway.

```
PE11-ATT-TRUNKPE11-ATT-TRUNK(sip-interface)# exit
PE11-ATT-TRUNK(session-router)#session-agent
PE11-ATT-TRUNK(session-agent)# hostname 207.242.225.210
PE11-ATT-TRUNK(session-agent)# port 5060
PE11-ATT-TRUNK(session-agent)# realm-id trunk-side
PE11-ATT-TRUNK(session-agent)#ping-method OPTIONS;hops=0
PE11-ATT-TRUNK(session-agent)#ping-interval 30
PE11-ATT-TRUNK(session-agent)#in-manipulationid changesendonly
PE11-ATT-TRUNK(session-agent)#rfc2833-payload 100
PE11-ATT-TRUNK(session-agent)# done
session-agent
        hostname                                207.242.225.210
        ip-address                              207.242.225.210
        realm-id                                trunk-side
        description                             ATT
        in-manipulationid                       changesendonly
        rfc2833-payload                         100
        refer-call-transfer                     enabled
```

You will now define the mediation server.

Defining Mediation Server 1

```
PE11-ATT-TRUNK(session-agent)# hostname medpool.partnersfb.com
PE11-ATT-TRUNK(session-agent)# port 5067
PE11-ATT-TRUNK(session-agent)# app-protocol sip
PE11-ATT-TRUNK(session-agent)# transport-method StaticTLS
PE11-ATT-TRUNK(session-agent)# realm-id core
PE11-ATT-TRUNK(session-agent)# ping-method OPTIONS;hops=0
PE11-ATT-TRUNK(session-agent)# ping-interval 30
PE11-ATT-Trunk(session-agent)# refer-call-transfer enabled
PE11-ATT-Trunk(session-agent)# out-translationid addplus1
PE11-ATT-Trunk(session-agent)# out-manipulationid checkFollowMeinDiversion
PE11-ATT-Trunk(session-agent)# load-balance-dns-query round-robin
PE11-ATT-TRUNK(session-agent)# done

session-agent
        hostname                                medpool.partnersfb.com
        port                                    5067
        transport-method                        StaticTLS
        realm-id                                core
        ping-method                             OPTIONS
        ping-interval                           30
        load-balance-dns-query                  round-robin
        out-translationid                       addplus1
        out-manipulationid                      checkFollowMeinDiversion
        refer-call-transfer                     enabled
```

10. Configure SIP routing

To configure the SIP routing, use the **local-policy** command under the **session-router** branch. To enter the **session-router** branch from **session-agent**, you issue the **exit** command, followed by the **local-policy** command.

We will first configure the route from the gateway to the mediation server.

```
PE11-ATT-TRUNK(session-agent)# exit
PE11-ATT-TRUNK(session-router)# local-policy
PE11-ATT-TRUNK(local-policy)# from-address *
PE11-ATT-TRUNK(local-policy)# to-address *
PE11-ATT-TRUNK(local-policy)# source-realm trunk-side
PE11-ATT-TRUNK(local-policy)# policy-attributes
PE11-ATT-TRUNK(local-policy-attributes)#next-hop medpool.partnersfb.com
PE11-ATT-TRUNK(local-policy-attributes)# realm core
PE11-ATT-TRUNK(local-policy-attributes)# done
PE11-ATT-TRUNK(local-policy-attributes)# exit
PE11-ATT-TRUNK(local-policy)# done
```

```
local-policy
        from-address                            *
        to-address                              *
        source-realm                            trunk-side
        policy-attribute
                next-hop                         medpool.partnersfb.com
                realm                                   core
```

We will now configure the route from the mediation server to the gateway.

```
PE11-ATT-TRUNK(local-policy)# from-address *
PE11-ATT-TRUNK(local-policy)# to-address *
PE11-ATT-TRUNK(local-policy)# source-realm core
PE11-ATT-TRUNK(local-policy)# policy-attributes
PE11-ATT-TRUNK(local-policy-attributes)# next-hop 207.242.225.210
PE11-ATT-TRUNK(local-policy-attributes)# realm trunk-side
PE11-ATT-TRUNK(local-policy-attributes)# app-protocol sip
PE11-ATT-TRUNK(local-policy-attributes)# done

PE11-ATT-TRUNK(local-policy-attributes)# exit
PE11-ATT-TRUNK(local-policy)# done

local-policy
        from-address                            *
        to-address                              *
        source-realm                            core
        policy-attribute
                next-hop                                207.242.225.210
                realm                                   trunk-side
                app-protocol                            SIP
```

11. Configure media handling

To configure the media handling, use the **steering-pool** command under the **media-manager** branch. To enter the **steering-pool** branch from **local-policy**, you issue the **exit** command twice, followed by the **media-manager** then the **steering-pool** command.

You will use the same IP address for the steering pool as the one used for the SIP interface. Note that the port ranges provide a means of limiting the number of concurrent media sessions within a given realm. For example, assigning 100 ports to a realm would limit it to 50 concurrent bidirectional calls, where two ports are assigned (one port for RTP and second port for RTCP).

```
PE11-ATT-TRUNK(local-policy)# exit
PE11-ATT-TRUNK(session-router)# exit
PE11-ATT-TRUNK(configure)# media-manager
PE11-ATT-TRUNK(media-manager)# steering-pool
PE11-ATT-TRUNK(steering-pool)# ip-address 192.168.4.130
```

```
PE11-ATT-TRUNK(steering-pool)# start-port 20000
PE11-ATT-TRUNK(steering-pool)# end-port 40000
PE11-ATT-TRUNK(steering-pool)# realm-id core
PE11-ATT-TRUNK(steering-pool)# done
steering-pool
        ip-address                              192.168.4.130
        start-port                              20000
        end-port                                40000
        realm-id                                core
```

You will now configure the media handling for the ATT realm.

```
PE11-ATT-TRUNK(steering-pool)# ip-address 155.212.214.181
PE11-ATT-TRUNK(steering-pool)# start-port 16384
PE11-ATT-TRUNK(steering-pool)# end-port 20000
PE11-ATT-TRUNK(steering-pool)# realm-id trunk-side
PE11-ATT-TRUNK(steering-pool)# done
steering-pool
        ip-address                              155.212.214.181
        start-port                              16384
        end-port                                20000
        realm-id                                trunk-side
```

12. SIP PRACK interworking and Media Handling

SIP PRACK Interworking

In order to establish an early media session for outbound calls, SFB gateway specification mandates the PSTN gateways to offer a reliable provisional response and for inbound calls offer INVITEs with a supported header The SBC can interwork and provide RFC 3262 PRACK interworking towards SFB and it is a mandatory configuration in all Oracle SBC – Microsoft SFB deployments. For this, the following need to be configured:

- Configure option 100rel-interworking on the sip-interface facing mediation server
- Configure a sip-feature to pass the 100rel in supported and require headers
- Configure a manipulation to add a Require:100rel header in incoming SIP INVITE from mediation server and delete the Supported:100rel header

```
PE11-ATT-TRUNK(session-router)# sip-interface
PE11-ATT-Trunk(sip-interface)# sel
<realm-id>:
1: core  192.168.4.130:5067
2: trunk-side 155.212.214.181:5060
selection: 1
PE11-ATT-TRUNK(sip-interface)#options 100rel-interworking
```

Configure Sip-feature to pass Supported and Require headers in SIP messages

```
PE11-ATT-TRUNK(session-router)#sip-feature
PE11-ATT-TRUNK(sip-feature)#name 100rel
PE11-ATT-TRUNK(sip-feature)#realm pstn
PE11-ATT-TRUNK(sip-feature)# support-mode-inbound Pass
PE11-ATT-TRUNK(sip-feature)# require-mode-inbound Pass
PE11-ATT-TRUNK(sip-feature)# proxy-require-mode-inbound Pass
PE11-ATT-TRUNK(sip-feature)# support-mode-outbound Pass
PE11-ATT-TRUNK(sip-feature)# require-mode-outbound Pass
PE11-ATT-TRUNK(sip-feature)# proxy-require-mode-outbound Pass
PE11-ATT-TRUNK(sip-feature)#done

sip-feature
      name                          100rel
      realm                         pstn
      support-mode-inbound          Pass
      require-mode-inbound          Pass
      proxy-require-mode-inbound    Pass
      support-mode-outbound         Pass
      require-mode-outbound         Pass
      proxy-require-mode-outbound   Pass
```

The manipulation to add Require:100rel header will be configured in the next section.

13. ESBC config for Microsoft Media Bypass feature

In order for Media Bypass to work, both Client and gateway (SBC) need to use the same RTP format, either SRTP (by default) or RTP. In default configuration of MS SFB, SFB client is required to use media encryption, so Media Bypass is mainly when media is encrypted (SRTP) and exchanged between SFB client and PSTN gateway (Net-Net ESBC).

Media Bypass from ESBC's perspective is routing RTP traffic to an endpoint/SFB client on a private routable network directly (instead of RTP going through the mediation server). To enable the SBC to handle the media bypass feature in SFB, you will need to set **restricted-latching** to **sdp** in the core realm (facing mediation server). Select the core realm from the **media-manager --- > realm-config** configuration branch.
Note: This setting is recommended irrespective of the media bypass setting.

```
PE11-ATT-Trunk(realm-config)#restricted-latching sdp
PE11-ATT-Trunk(realm-config)#done
      realm-config
```

```
identifier                          core
network-interfaces                  s1p0:0
mm-in-realm                         enabled
media-sec-policy                    sdespolicy
restricted-latching                 sdp
refer-call-transfer                 enabled
codec-policy                        TrunkCodecs
```

Recently, in some accounts where MS Lync and Oracle SBCs are deployed for enterprise voice and SIP trunk termination to an enterprise, there have been complaints of the PSTN caller hearing a silence when a call is placed from PSTN to a SFB user on the enterprise especially when Media Bypass is enabled on MS SFB
The configuration note below aims to explain this scenario briefly, steps taken to rectify this issue and proposed workaround by Acme Packet. The workaround is an interim solution while a permanent solution is being researched and developed by Oracle Communications Engineering.

Media Bypass

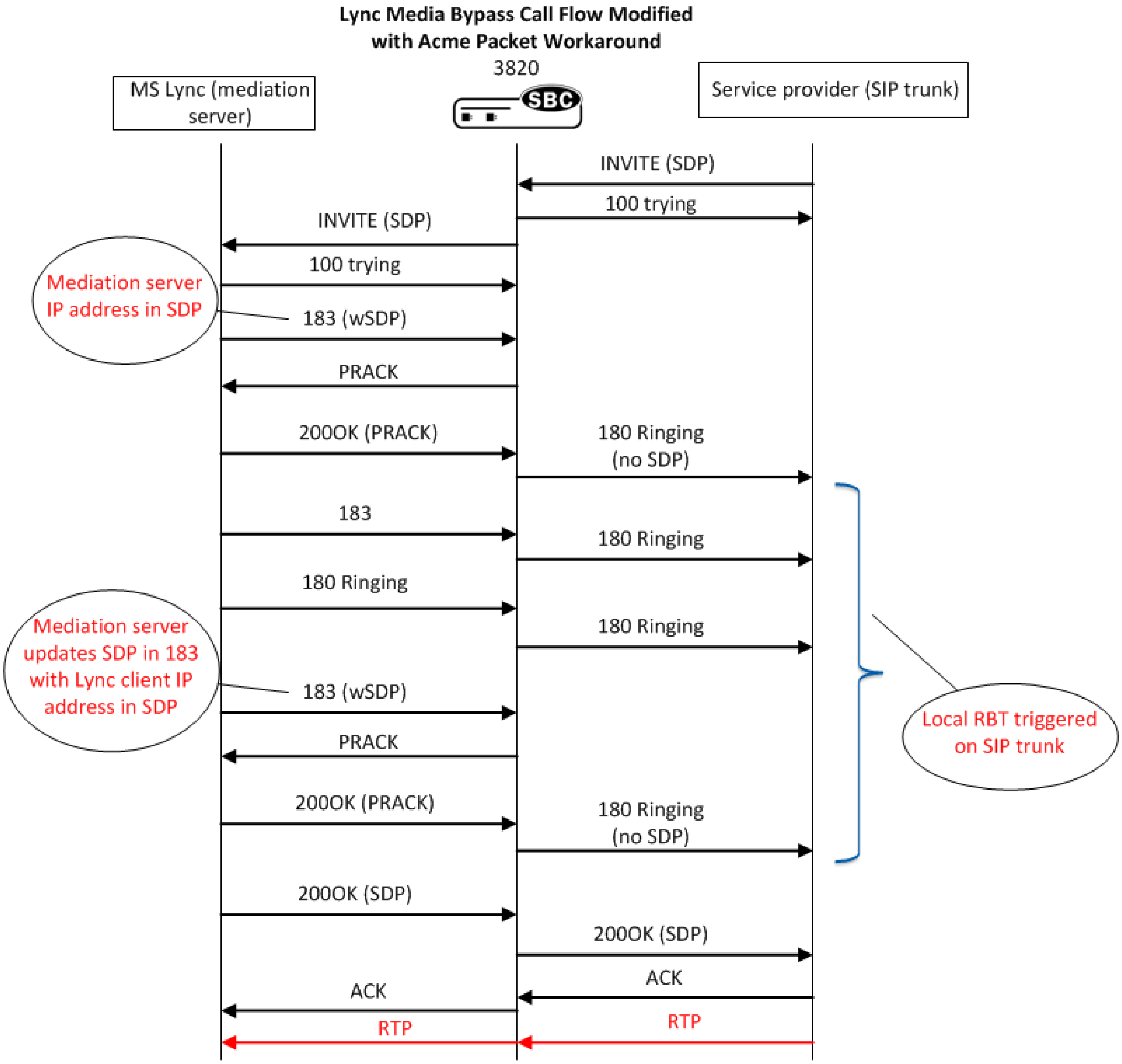
As explained earlier in the document, in order for Media Bypass to work, both Client and gateway (SBC) need to use the same RTP format, either SRTP (by default) or RTP. In default configuration of MS SFB, SFB client is required to use media encryption, so Media Bypass is mainly when media is encrypted (SRTP) and exchanged between SFB client and PSTN gateway (E-SBC).

Signaling between mediation server and SBC is a little different (Two 183s with SDP coming from mediation server) when media bypass is enabled on Lync.

The following is the call flow:

After signaling 183 with SDP, SFB never plays any early media and expects gateway (E-SBC) to signal appropriately to the SIP Trunk provider to follow RFC 3960 and play local RBT. The second 183w SDP coming from Mediation server which is forwarded to the SIP trunk and stops the local RBT which was started after 180 Ringing was sent, hence PSTN caller would hear a silence before Lync client answers call.

The solution here is to present 180 ringing (i.e. convert all 183s on lync side to 180 ringing towards SIP trunk and strip the SDP) to trigger RBT in ISUP. The call flow is modified with the help of Oracle Communication's robust Sip Manipulation and Sip Response Map features to the following:



The following header rules needs to be included in the manipulation that is applied on the realm or sip-interface facing Lync to modify the signaling traffic sent from Lync.

```
        header-rule
                name                           delsupported
                header-name                    Supported
                action                         delete
                comparison-type                case-sensitive
                msg-type                       request
                methods                        INVITE
                match-value
                new-value
        header-rule
                name                           addrequireinINVITE
                header-name                    Require
                action                         add
                comparison-type                case-sensitive
                msg-type                       request
```

```
                methods                         INVITE
                match-value
                new-value                       100rel
        header-rule
                name                            formod183
                header-name                     From
                action                          sip-manip
                comparison-type                 case-sensitive
                msg-type                        any
                methods
                match-value
                new-value                       Stripsdp183 (the manipulation
Stripsdp183 is mentioned below)
```

```
sip-manipulation
        name                            Stripsdp183
        description                     For incoming 183 from Lync, strip SDP
        split-headers
        join-headers
        header-rule
                name                            check183
                header-name                     @status-line
                action                          store
                comparison-type                 pattern-rule
                msg-type                        any
                methods
                match-value
                new-value
                element-rule
                        name                            is183
                        parameter-name
                        type                            status-code
                        action                          store
                        match-val-type                  any
                        comparison-type                 pattern-rule
                        match-value                     183
                        new-value
        header-rule
                name                            delSDP
                header-name                     Content-Type
                action                          manipulate
                comparison-type                 case-insensitive
                msg-type                        any
                methods
                match-value                     $check183.$is183
                new-value
                element-rule
                        name                            del183SDP
                        parameter-name                  application/sdp
                        type                            mime
                        action                          delete-element
                        match-val-type                  any
                        comparison-type                 boolean
                        match-value
                        new-value
        header-rule
                name                            delContentType
```

```
                header-name                     Content-Type
                action                          manipulate
                comparison-type                 boolean
                msg-type                        any
                methods
                match-value                     $check183.$is183
                new-value
                element-rule
                        name                            delCT
                        parameter-name                  *
                        type                            header-param
                        action                          delete-header
                        match-val-type                  any
                        comparison-type                 case-sensitive
                        match-value
                        new-value
```

The following sip response map needs to be configured and applied on the sip interface facing ATT.

```
response-map
        last-modified-by                admin@10.0.221.18
        last-modified-date              2012-06-04 11:14:17
        name                            change183to180
        entries
                                        183 -> 180 (Ringing)
      sip-interface
              state                           enabled
              realm-id                        ATT
              description
              sip-port
                      address                         192.20.0.108
                      port                            5060
                      transport-protocol              UDP
                      tls-profile
                      multi-home-addrs
                      allow-anonymous                 agents-only
                      ims-aka-profile
                    ....

             response-map                    change183to180
```

14. Configure Sip-manipulations and translation rules

In order to cater to AT&T's and SFB's call flow standards, we need to configure certain header manipulation rules (HMR). The **sip-manipulation** element can be found under the **session-router** element.

The HMR applied to the signaling towards the trunk performs the following changes:

- The Request-URI is modified to include the ip address and port of the trunk device
- The uri-host portion of the From header is replaced with the FQDN of the trunk, in our case the uri-host is changed to IP Address of the SBC facing the AT&T trunk

- In the Contact header, we have header rules to strip +1 from the uri-user and replace the uri-host and uri-port portions with the SBC's local ip and port of the interface facing the trunk.
- In the Route header we remove the +1 from the uri-user.
- For privacy enabled calls, SFB sends the phone number in the From header. It indicates that it is a privacy enabled calls using the 'Privacy:id' header. For such calls, we replace the phone number in the uri-user of the From header with 'anonymous'.

To conform SFB's signaling per the trunk's specification, we modify the messages coming from SFB and also make some changes to messages before they are sent to SFB.

The following changes are applied to the messages coming from SFB:

- We add a 'Require:100rel' header in incoming SIP INVITE from mediation server and delete the 'Supported:100rel' header as mentioned in the SIP PRACK interworking section.
- To enabled ringback on transfers, we replace the 'a=inactive' line in SDP of the INVITEs with 'a=sendonly'. For more information, please refer to the Ring-back tone during Transfers section.

To the messages sent to SFB, the following changes are applied:

- The uri-hosts of the From and To headers are replaced with SBC's local ip and SFB's ip.
- In the From and To headers we remove the +1 from the uri-user, when the uri-user is anonymous.
- At last we have a rule to insert +1 in the uri-user of the Contact header as SFB server is configured for E.164 format.

Here is the list of HMR's used:

- ChangeContact - Fixes the contact header offered by SFB before the message is sent to Trunk
- Changeinactosendonly - SBC changes SDP from inactive to sendonly on INVITEs for hold (required to trigger audio playback from SIP Trunk)
- Check183 - Check the response to INVITE is 183 session progress
- NATting - NAT From & To header with correct IP information
- convert183to180 - Convert 183 to 180 for triggering early media
- ForEarlyMedia - To locally handle PRACK interworking
- Lyncprivacy - NAT plus recvonly to inactive
- ChangeforPAIandNAT - configured on the trunk side to change the Privacy, Nating

A manipulation, ChangeContact, will need to be configured to change the format of the CONTACT header which will then be referenced in the manipulation that is finally applied to the realm or sip-interface facing AT&T.

The manipulation consists of two header rules – StoreFromnumber and ChangeContact. The StoreFromnumber header rule stores the uri-user-only element in the From header which is then added as the uri-user in the Contact header in the ChangeContact header rule.

```
sip-manipulation
        name                            ChangeContact
        description
        split-headers
        join-headers
        header-rule
                name                            StoreFromnumber
```

```
                header-name                    From
                action                         manipulate
                comparison-type                case-sensitive
                msg-type                       any
                methods
                match-value
                new-value
                element-rule
                        name                         StoreFromnumber_er
                        parameter-name
                        type                           uri-user-only
                        action                         store
                        match-val-type                 any
                        comparison-type                case-sensitive
                        match-value
                        new-value
        header-rule
                name                           ChangeContact
                header-name                    Contact
                action                         manipulate
                comparison-type                case-sensitive
                msg-type                       any
                methods
                match-value
                new-value
                element-rule
                        name                           ChangeContact_er
                        parameter-name
                        type                           uri-user
                        action                         add
                        match-val-type                 any
                        comparison-type                case-sensitive
                        match-value
                        new-value
$StoreFromnumber.$StoreFromnumber_er.$0
```

The manipulation ChangeforPAIandNAT is configured on the trunk side to change the Privacy, Nating.

```
sip-manipulation
        name                           ChangeforPAIandNAT
        description                    Change PAI and NATing
        split-headers
        join-headers
        header-rule
                name                           forprivacy
                header-name                    From
                action                         sip-manip
                comparison-type                case-sensitive
                msg-type                       any
                methods
                match-value
                new-value                      NATting
        header-rule
                name                           fordiv
```

```
            header-name                    From
            action                         sip-manip
            comparison-type                case-sensitive
            msg-type                       any
            methods
            match-value
            new-value                      AddDiversion
    header-rule
            name                           ForREFER
            header-name                    From
            action                         sip-manip
            comparison-type                case-sensitive
            msg-type                       any
            methods
            match-value
            new-value                      changeRefer
    header-rule
            name                           ForREFER
            header-name                    From
            action                         sip-manip
            comparison-type                case-sensitive
            msg-type                       any
            methods
            match-value
            new-value                      ChangeContact
    header-rule
            name                           Refer_header
            header-name                    Referred-By
            action                         manipulate
            comparison-type                case-sensitive
            msg-type                       any
            methods
            match-value
            new-value
            element-rule
                    name                           referredbyhdr
                    parameter-name
                    type                           uri-host
                    action                         replace
                    match-val-type                 any
                    comparison-type                case-sensitive
                    match-value
                    new-value                      $LOCAL_IP
    header-rule
            name                           changePrivacy
            header-name                    From
            action                         sip-manip
            comparison-type                case-sensitive
            msg-type                       any
            methods
            match-value
            new-value                      Check_privacy_header
```

The sip-manipulation then needs to be applied on the realm or sip-interface or session-agent towards the ATT trunk side. We apply it on the sip-interface here:

```
PE11-ATT-TRUNK(session-router)# sip-interface
PE11-ATT-Trunk(sip-interface)# sel
<realm-id>:
1: core  192.168.4.130:5067
2: trunk-side 155.212.214.181:5060

selection: 2
PE11-ATT-Trunk(sip-interface)# out-manipulationid ChangeforPAIandNAT
PE11-ATT-Trunk(sip-interface)# done
```

In order to complete the calls successfully per AT&T's signaling specifications, we need to configure manipulation rules on the realm facing SFB. The manipulations are mentioned below.
The sip-manipulation NATting ensure topology hiding.

```
sip-manipulation
        name                          NATting
        description
        split-headers
        join-headers
        header-rule
                name                          From
                header-name                   From
                action                        manipulate
                comparison-type               case-sensitive
                msg-type                      any
                methods
                match-value
                new-value
                element-rule
                        name                          From_header
                        parameter-name
                        type                          uri-host
                        action                        replace
                        match-val-type                any
                        comparison-type               case-sensitive
                        match-value
                        new-value                     $LOCAL_IP
        header-rule
                name                          To
                header-name                   To
                action                        manipulate
                comparison-type               case-sensitive
                msg-type                      request
                methods
                match-value
                new-value
                element-rule
                        name                          To
                        parameter-name
```

```
                        type                          uri-host
                        action                        replace
                        match-val-type                any
                        comparison-type               case-sensitive
                        match-value
                        new-value                     $REMOTE_IP
```

For simultaneous ringing, the following manipulation is configured

```
sip-manipulation
        name                          ATT-Simulring
        description                   HMR for simul ring towards Lync
        split-headers
        join-headers
        header-rule
                name                          getTo
                header-name                   To
                action                        store
                comparison-type               case-sensitive
                msg-type                      request
                methods                       INVITE
                match-value
                new-value
                element-rule
                        name                          getTag
                        parameter-name                tag
                        type                          header-param
                        action                        store
                        match-val-type                any
                        comparison-type               pattern-rule
                        match-value
                        new-value
        header-rule
                name                          checkHoldSdp
                header-name                   Content-Type
                action                        store
                comparison-type               boolean
                msg-type                      request
                methods                       INVITE
                match-value                   !$getTo.$getTag
                new-value
                element-rule
                        name                          checkIP
                        parameter-name                application/sdp
                        type                          mime
                        action                        store
                        match-val-type                any
                        comparison-type               case-sensitive
```

```
                        match-value                        \Rc=IN IP4 0\.0\.0\.0\b
                        new-value
        header-rule
                name                               fixSdptest
                header-name                        Content-Type
                action                             manipulate
                comparison-type                    boolean
                msg-type                           request
                methods                            INVITE
                match-value                        $checkHoldSdp.$checkIP
                new-value
                element-rule
                        name                               replaceIP
                        parameter-name                     application/sdp
                        type                               mime
                        action                             find-replace-all
                        match-val-type                     any
                        comparison-type                    pattern-rule
                        match-value                        \Rc=IN IP4
(0\.0\.0\.0)\b[[:1:]]
                        new-value                          $LOCAL_IP
        header-rule
                name                               checkmodinactive
                header-name                        Content-Type
                action                             store
                comparison-type                    boolean
                msg-type                           request
                methods                            INVITE
                match-value                        !$getTo.$getTag
                new-value
                element-rule
                        name                               checkstate
                        parameter-name                     application/sdp
                        type                               mime
                        action                             store
                        match-val-type                     any
                        comparison-type                    pattern-rule
                        match-value                        \Ra=inactive\b
                        new-value
        header-rule
                name                               fixinactive
                header-name                        Content-Type
                action                             manipulate
                comparison-type                    boolean
                msg-type                           request
                methods                            INVITE
                match-value                        $checkmodinactive.$checkstate
                new-value
                element-rule
                        name                               replaceAttribute
```

```
                        parameter-name                 application/sdp
                        type                           mime
                        action                         find-replace-all
                        match-val-type                 any
                        comparison-type                pattern-rule
                        match-value                    \Ra=inactive\b
                        new-value
```

The manipulations NATting and ATT-Simulring need to be applied to manipulate the signaling sent to devices in the realm core. Hence the following nested sip-manipulation Lyncprivacy is configured.

```
sip-manipulation
        name                           Lyncprivacy
        description                    NAT plus recvonly to inactive
        split-headers
        join-headers
        header-rule
                name                           doNATforlync
                header-name                    From
                action                         sip-manip
                comparison-type                case-sensitive
                msg-type                       any
                methods
                match-value
                new-value                      NATting
        header-rule
                name                           manipPPreferredIdentity
                header-name                    P-Preferred-Identity
                action                         manipulate
                comparison-type                case-sensitive
                msg-type                       request
                methods
                match-value
                new-value
                element-rule
                        name                     PPreferredIdentityURIHost
                        parameter-name
                        type                           uri-host
                        action                         replace
                        match-val-type                 any
                        comparison-type                case-sensitive
                        match-value
                        new-value                      $LOCAL_IP
        header-rule
                name                           simulring
                header-name                    From
                action                         sip-manip
                comparison-type                case-sensitive
```

```
                msg-type                        request
                methods                         INVITE
                match-value
                new-value                       ATT-Simulring
```

This manipulation is applied on the sip-interface or realm facing SFB.

```
PE11-ATT-TRUNK(session-router)# sip-interface
PE11-ATT-Trunk(sip-interface)# sel
<realm-id>:
1: MS-Lync-Peer  192.168.2.130:5068
Note:2: ATT 192.20.0.108:5060

selection: 1
PE11-ATT-Trunk(sip-interface)# out-manipulationid Lyncprivacy
PE11-ATT-Trunk(sip-interface)# done
```

During call transfer to a PSTN party, the transfer completes but the calling party does not hear a ring back tone during the process of transfer. The INVITE Lync sends to the SBC to initiate the transfer contains the SDP attribute, a=inactive which is forwarded to the trunk and as a result of which the SBC cannot play the ring back tone to the original PSTN caller (while call is being transferred). A sendonly attribute is required for MoH and transfer scenarios for the calling party to be able to hear ringback or MoH when it is kept on hold. The SBC is able to signal appropriately towards the SIP trunk by changing the a=inactive SDP attribute in the INVITE to sendonly towards PSTN. This attribute needs to be changed to a=sendrecv when it is sent to AT&T so that the ringback tone or the MOH can be heard.

Sip manipulations are configured to make the necessary changes. The manipulation Changeinactosendonly is configured to change the SDP attribute from a=inactive to a=sendonly in the INVITEs sent to the calling party for transfer.

```
   sip-manipulation
           name                            Changeinactosendonly
           description                     Change inactive to sendonly for pstn tran
           split-headers
           join-headers
           header-rule
                   name                            changeSDP
                   header-name                     Content-Type
                   action                          manipulate
                   comparison-type                 case-sensitive
                   msg-type                        request
                   methods                         INVITE
                   match-value
                   new-value
                   element-rule
                           name                            inacttosendonly
                           parameter-name                  application/sdp
                           type                            mime
                           action                          find-replace-all
                           match-val-type                  any
                           comparison-type                 pattern-rule
```

```
                    match-value                    a=inactive
                    new-value                      a=sendonly
```

Note:
To change the a=sendonly to a=sendrecv before sending the INVITE to AT&T, we have a header rule Changesendonlytosendrecv included in the manipulation Privacy that is applied on the sip-interface facing AT&T.
A nested sip manipulation Forearlymedia is configured to include the header rules mentioned in the section "SIP PRACK interworking and Media Handling" and the manipulation Changeinactosendonly

The manipulation ChangeforPAIandNAT is configured on the trunk side to change the Privacy, Nating.

```
sip-manipulation
        name                          ChangeforPAIandNAT
        description                   Change PAI and NATing
        split-headers
        join-headers
        header-rule
                name                          forprivacy
                header-name                   From
                action                        sip-manip
                comparison-type               case-sensitive
                msg-type                      any
                methods
                match-value
                new-value                     NATting
        header-rule
                name                          fordiv
                header-name                   From
                action                        sip-manip
                comparison-type               case-sensitive
                msg-type                      any
                methods
                match-value
                new-value                     AddDiversion
        header-rule
                name                          ForREFER
                header-name                   From
                action                        sip-manip
                comparison-type               case-sensitive
                msg-type                      any
                methods
                match-value
                new-value                     changeRefer
        header-rule
                name                          ForREFER
                header-name                   From
                action                        sip-manip
                comparison-type               case-sensitive
                msg-type                      any
                methods
                match-value
                new-value                     ChangeContact
        header-rule
                name                          Refer_header
                header-name                   Referred-By
```

```
                    action                              manipulate
                    comparison-type                     case-sensitive
                    msg-type                            any
                    methods
                    match-value
                    new-value
                    element-rule
                            name                                referredbyhdr
                            parameter-name
                            type                                uri-host
                            action                              replace
                            match-val-type                      any
                            comparison-type                     case-sensitive
                            match-value
                            new-value                           $LOCAL_IP
            header-rule
                    name                                changePrivacy
                    header-name                         From
                    action                              sip-manip
                    comparison-type                     case-sensitive
                    msg-type                            any
                    methods
                    match-value
                    new-value                           Check_privacy_header
```

Sip manipulations for checking privacy header.

```
    sip-manipulation
            name                                Check_privacy_header
            description                         Check for privacy and overwrite FROM
            split-headers
            join-headers
            header-rule
                    name                                ChechForPrivacy
                    header-name                         Privacy
                    action                              manipulate
                    comparison-type                     case-sensitive
                    msg-type                            request
                    methods                             INVITE
                    match-value
                    new-value
            header-rule
                    name                                OverwriteFrom
                    header-name                         From
                    action                              manipulate
                    comparison-type                     boolean
                    msg-type                            request
                    methods                             INVITE
                    match-value                         $ChechForPrivacy
                    new-value
                    element-rule
                            name                                OverwriteUser
                            parameter-name
                            type                                uri-user
```

```
                            action                              find-replace-all
                            match-val-type                      any
                            comparison-type                     case-sensitive
                            match-value
                            new-value                           anonymous
                header-rule
                        name                            remove_P_Asserted_ID
                        header-name                     P-ASSERTED-IDENTITY
                        action                          delete
                        comparison-type                 case-insensitive
                        msg-type                        request
                        methods                         INVITE
                        match-value
                        new-value
                header-rule
                        name                            OverwriteFromDisplay
                        header-name                     From
                        action                          manipulate
                        comparison-type                 boolean
                        msg-type                        request
                        methods                         INVITE
                        match-value                     $ChechForPrivacy
                        new-value
                        element-rule
                            name                                OverwriteDisplay
                            parameter-name
                            type                                uri-display
                            action                              find-replace-all
                            match-val-type                      any
                            comparison-type                     case-sensitive
                            match-value
                            new-value                           "\"Anonymous\" "
                header-rule
                        name                            add_P_Asserted_ID
                        header-name                     P-Asserted-Identity
                        action                          add
                        comparison-type                 case-sensitive
                        msg-type                        request
                        methods                         INVITE
                        match-value
                        new-value                       "\"Unavailable\" <sip:$uri-user
        @Anonymous.Invalid>"
```

Following HMR (checkFollowMeinDiversion) is required when sequential-ring feature is enabled for particular DIDs. In a Sequential ring call flow – a trunk user first calls number 1 – since Sequential ring is enabled for this DID and no answer response was received by the trunk – trunk sends a new INVITE with sendonly in the original SDP offer to the second DID (which is configured as the sequential ring DID). To the SFB's mediation server the second INVITE with sendonly is not an acceptable offer – the mediation server rejects the INVITE if the first INVITE has nothing other than sendrecv. In order to work around this limitation on SFB, following HMR was built.

A header rule adds the text "Follow me" to from header's uri-display if diversion header with follow-me value is present. If the INVITE message doesn't have the Diversion header, the HMR will delete "Follow Me" from uri-display.

Before sending the 200 OK – the SBC checks if the “Follow Me” is present in the uri-display – only if present will add “recvonly” if not, allows the 200 OK without any manipulation.

```
sip-manipulation
        name                           checkFollowMeinDiversion
        description                    check for follow me in diversion header
        split-headers
        join-headers
        header-rule
                name                           checkFollowMe
                header-name                    Diversion
                action                         manipulate
                comparison-type                case-sensitive
                msg-type                       request
                methods                        INVITE
                match-value
                new-value
                element-rule
                        name                           checkFollowMe_er
                        parameter-name                 reason
                        type                           header-param
                        action                         store
                        match-val-type                 any
                        comparison-type                case-sensitive
                        match-value                    follow-me
                        new-value
        header-rule
                name                           addFollowMeFrom
                header-name                    From
                action                         manipulate
                comparison-type                boolean
                msg-type                       request
                methods                        INVITE
                match-value                    $checkFollowMe.$checkFollowMe_er
                new-value
                element-rule
                        name                           addFollowMeFrom_er
                        parameter-name
                        type                           uri-display
                        action                         add
                        match-val-type                 any
                        comparison-type                case-sensitive
                        match-value
                        new-value                      "Follow Me"
        header-rule
                name                           checkDiv
                header-name                    Diversion
                action                         manipulate
                comparison-type                case-sensitive
                msg-type                       request
                methods                        INVITE
                match-value
                new-value
        header-rule
                name                           removeFollowme
                header-name                    From
                action                         manipulate
```

```
                comparison-type                 boolean
                msg-type                        request
                methods                         INVITE
                match-value                     !$checkDiv
                new-value
                element-rule
                        name                            removeFollowme_er
                        parameter-name
                        type                            uri-display
                        action                          find-replace-all
                        match-val-type                  any
                        comparison-type                 case-sensitive
                        match-value                     Follow Me
                        new-value
        header-rule
                name                            checkFollowMeFrom200
                header-name                     From
                action                          manipulate
                comparison-type                 case-sensitive
                msg-type                        reply
                methods
                match-value
                new-value
                element-rule
                        name                            checkFollowMeFrom200_er
                        parameter-name
                        type                            uri-display
                        action                          store
                        match-val-type                  any
                        comparison-type                 case-sensitive
                        match-value                     Follow Me
                        new-value
        header-rule
                name                            addrecvOnly
                header-name                     Content-Type
                action                          manipulate
                comparison-type                 boolean
                msg-type                        reply
                methods
                match-value
   $checkFollowMeFrom200.$checkFollowMeFrom200_er
                new-value
                element-rule
                        name                            addrecvOnly_er
                        parameter-name                  application/sdp
                        type                            mime
                        action                          find-replace-all
                        match-val-type                  any
                        comparison-type                 pattern-rule
                        match-value
                        new-value
   $ORIGINAL+$CRLF+"a=recvonly"
```

```
sip-manipulation
        name                           Forearlymedia
        description
        split-headers
        join-headers
        header-rule
                name                           delsupported
                header-name                    Supported
                action                         delete
                comparison-type                case-sensitive
                msg-type                       request
                methods                        INVITE
                match-value
                new-value
        header-rule
                name                           addrequireinINVITE
                header-name                    Require
                action                         add
                comparison-type                case-sensitive
                msg-type                       request
                methods                        INVITE
                match-value
                new-value                      100rel
        header-rule
                name                           mod183
                header-name                    From
                action                         sip-manip
                comparison-type                case-sensitive
                msg-type                       any
                methods
                match-value
                new-value                      Stripsdp183
        header-rule
                name                           inactosendonly
                header-name                    From
                action                         sip-manip
                comparison-type                case-sensitive
                msg-type                       request
                methods
                match-value
                new-value                      Changeinactosendonly
        header-rule
                name                           CheckFollowme
                header-name                    From
                action                         sip-manip
                comparison-type                case-sensitive
                msg-type                       request
                methods
                match-value
                new-value                      checkFollowMeinDiversion
```

The sip-interface or realm facing Lync is configured with this manipulation as the in-manipulationid.

```
PE11-ATT-TRUNK(session-router)# sip-interface
PE11-ATT-Trunk(sip-interface)# sel
<realm-id>:
1: core  192.168.4.130:5067
2: trunk-side 155.212.214.181:5060

selection: 1
PE11-ATT-Trunk(sip-interface)# in-manipulationid Forearlymedia
PE11-ATT-Trunk(sip-interface)# done
```

15. Verify configuration integrity

You will verify your configuration referential integrity before saving and activating it with the **verify-config** command. This command is available from Superuser Mode. To enter the Superuser Mode from steering-pool, you issue the exit command three times.

```
PE11-ATT-TRUNK(configure)# exit
PE11-ATT-TRUNK# verify-config
-----------------------------------------------------------------------

Verification successful! No errors nor warnings in the configuration
```

16. Save and activate your configuration

You will now save your configuration with the **`save-config`** command. This will make it persistent through reboots, but it will not take effect until after you issue the **`activate-config`** command.

```
PE11-ATT-TRUNK# save-config
checking configuration
Save-Config received, processing.
waiting for request to finish
Request to 'SAVE-CONFIG' has Finished,
Save complete
Currently active and saved configurations do not match!
To sync & activate, run 'activate-config' or 'reboot activate'.

PE11-ATT-TRUNK# activate-config
Activate-Config received, processing.
waiting for request to finish
Setting phy0 on Slot=0, Port=0, MAC=00:08:25:03:FC:43,
VMAC=00:08:25:03:FC:43
Setting phy1 on Slot=1, Port=0, MAC=00:08:25:03:FC:45,
VMAC=00:08:25:03:FC:45
Request to 'ACTIVATE-CONFIG' has Finished,
Activate Complete
```

Phase 2 – Configuring the Skype for Business Server

The enterprise will have a fully functioning Skype for Business Server infrastructure with Enterprise Voice deployed and a Mediation Server dedicated to this installation. If there is no Mediation Server present for this purpose, one will have to be deployed.

There are two parts for configuring Lync Server to operate with the Oracle ESBC:

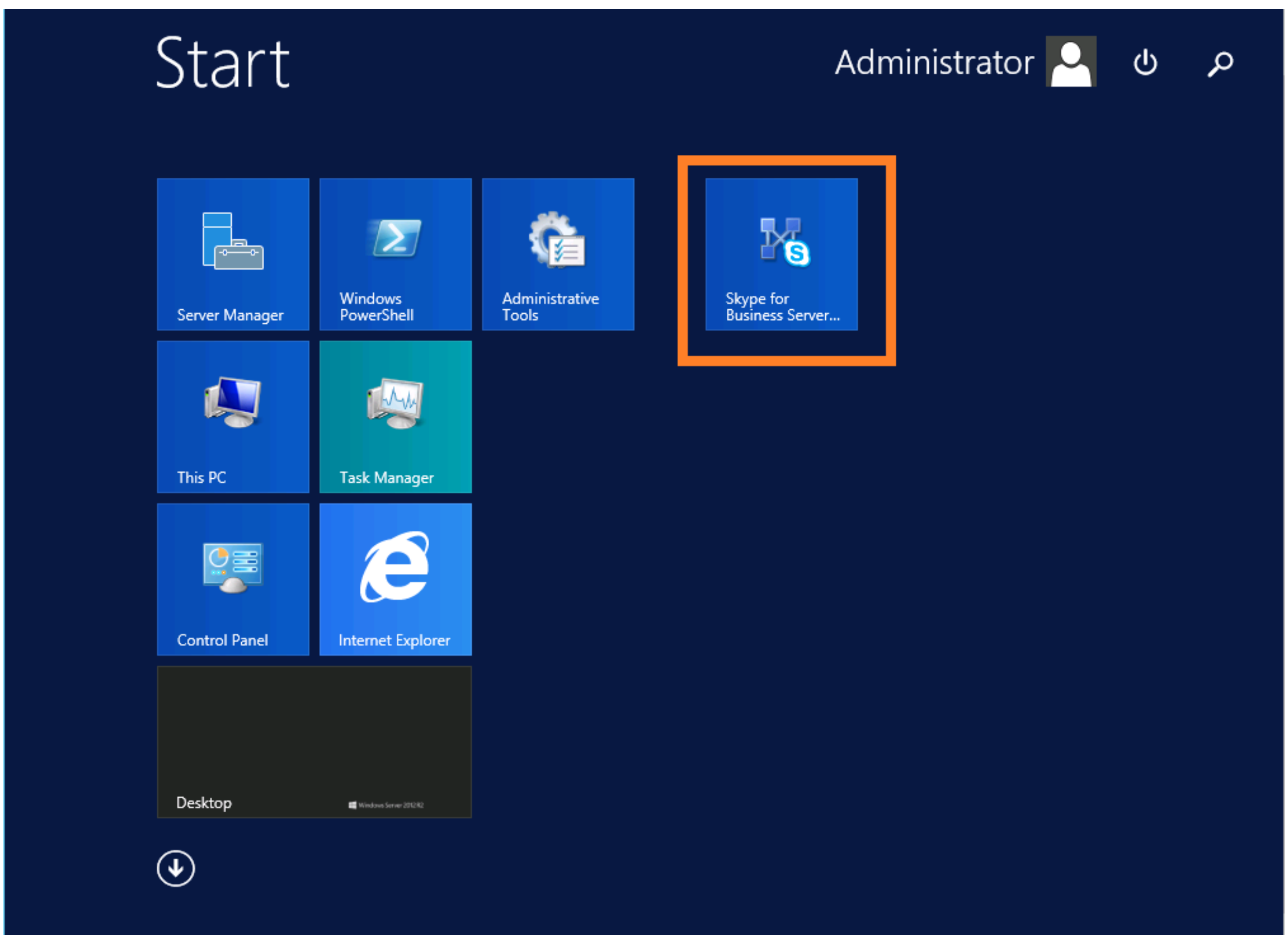
- Adding the Net-Net SBC as a PSTN gateway to the SFB Server infrastructure
- Creating a route within the SFB Server infrastructure to utilize the SIP trunk connected to the SBC.

To add the PSTN gateway, we will need:

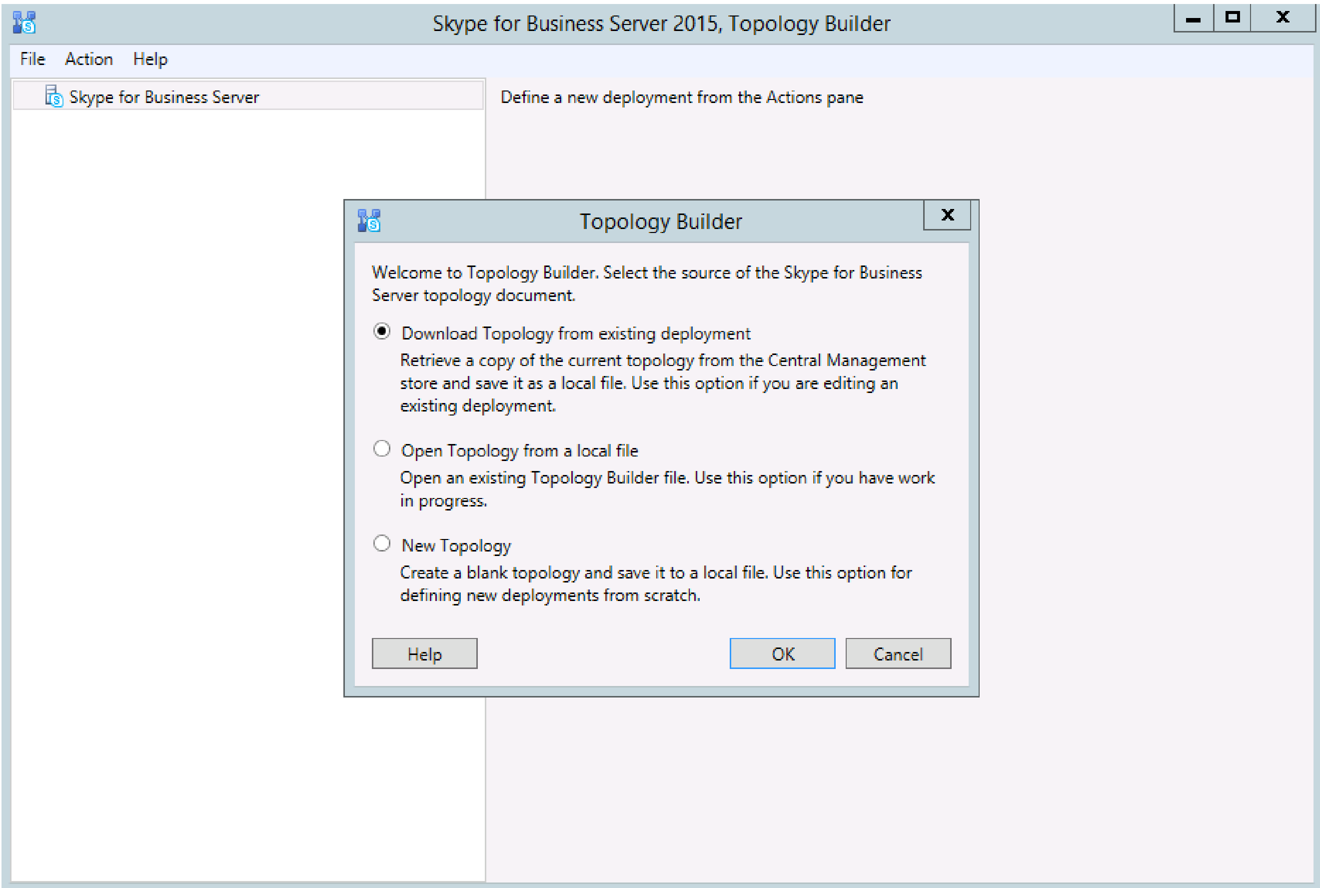
- IP addresses of the external facing NICs of the Mediation Servers
- IP address of the Net-Net SBC external facing port
- Rights to administer Lync Server Topology Builder
- Access to the SFB Server Topology Builder

The following process details the steps to add PSTN gateway

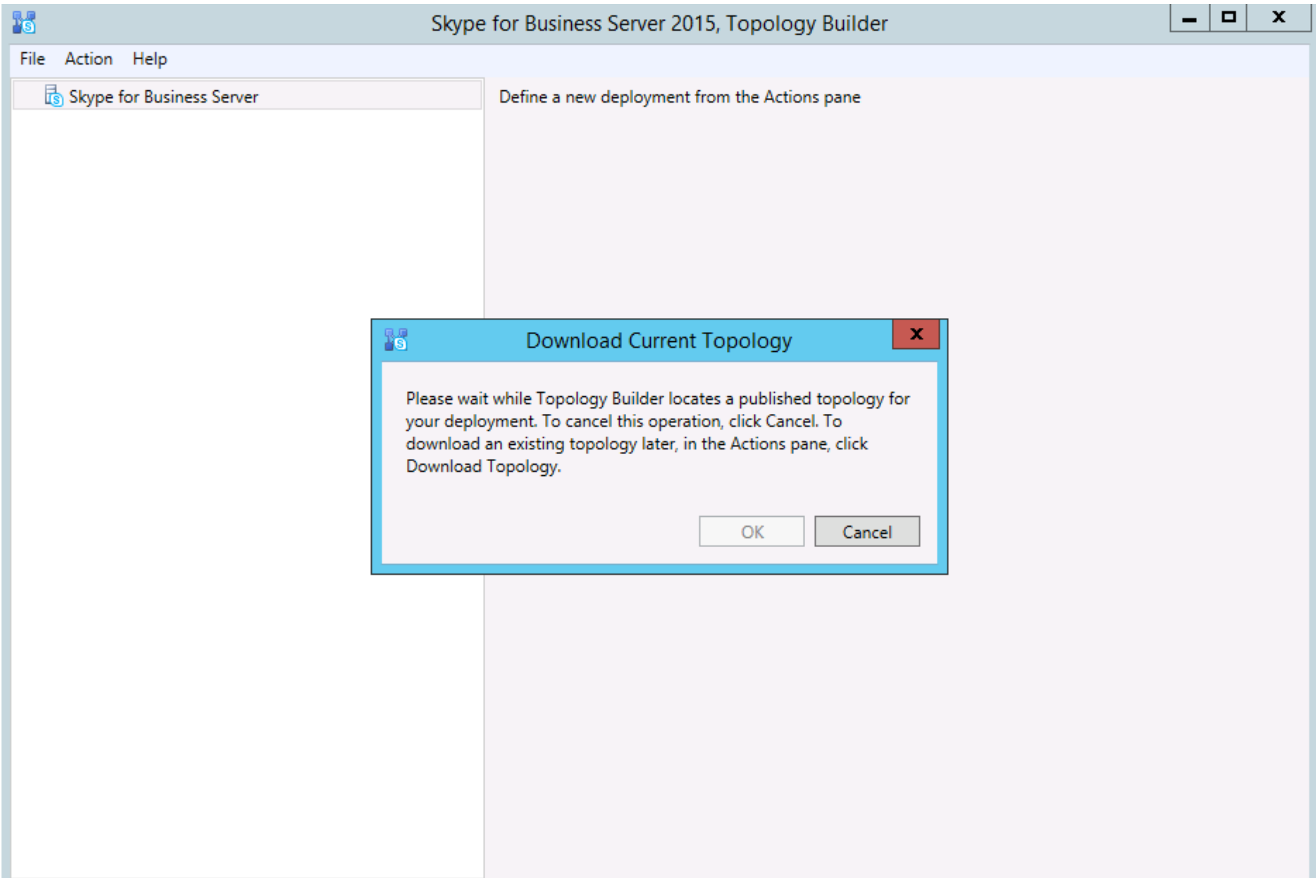
1. On the server where the Topology Builder is located start the console.
2. From the Start bar, select SFB Server Topology Builder.



3. The Topology Builder window will now be displayed. Select Download Topology from existing deployment.

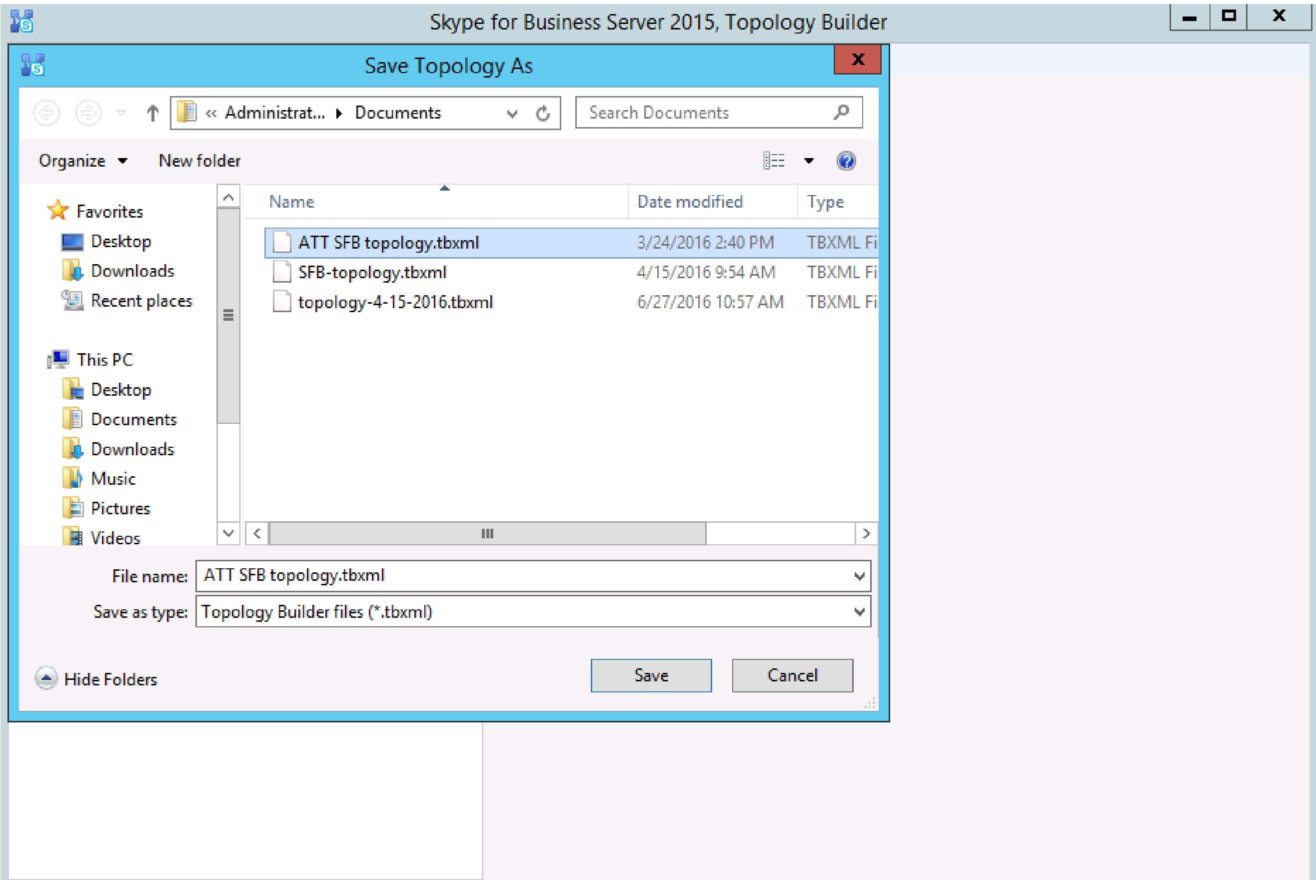


4. You will then see a screen showing that the current toplogy is being downloaded. Click the Ok button.

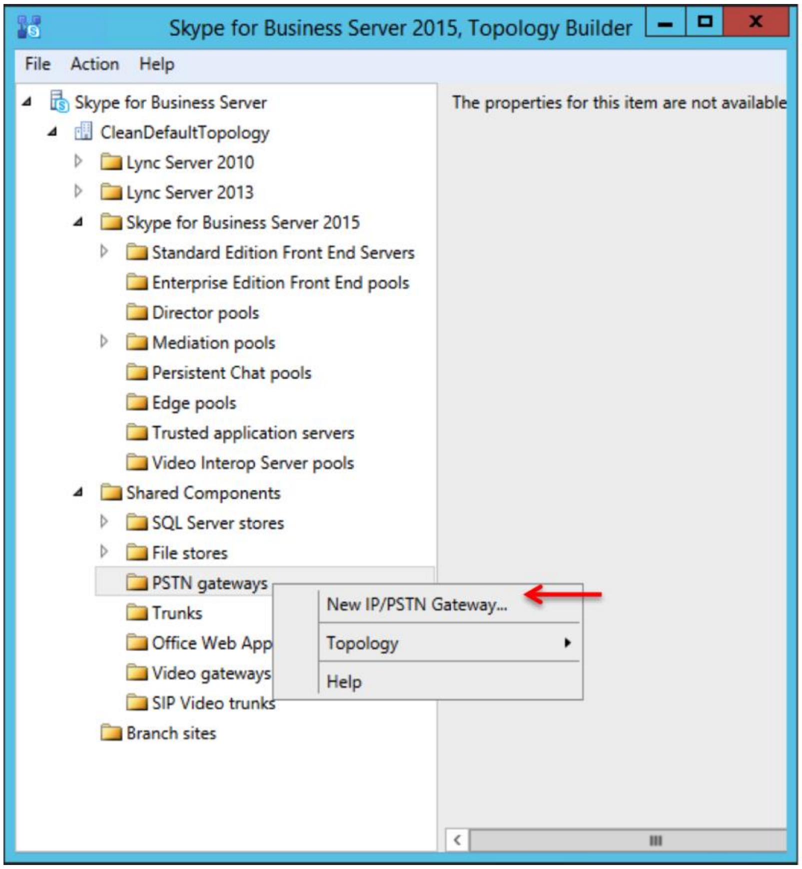


5. Next you will be prompted to save the topology which you have imported. You should revision the name or number of the topology according to the standards used within the enterprise. Click the Save button

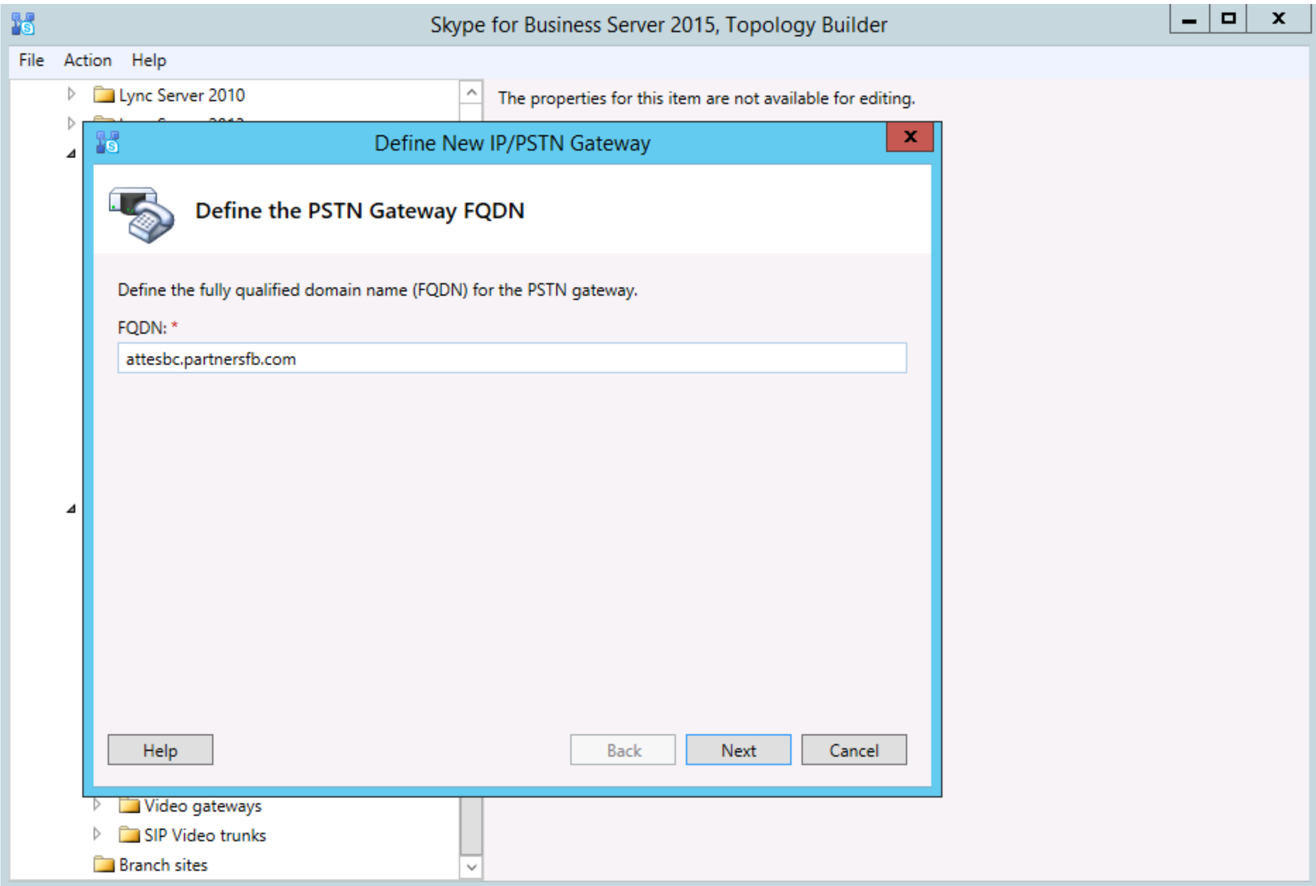
Note: This keeps track of topology changes and, if desired, will allow you to fall back from any changes you make during this installation



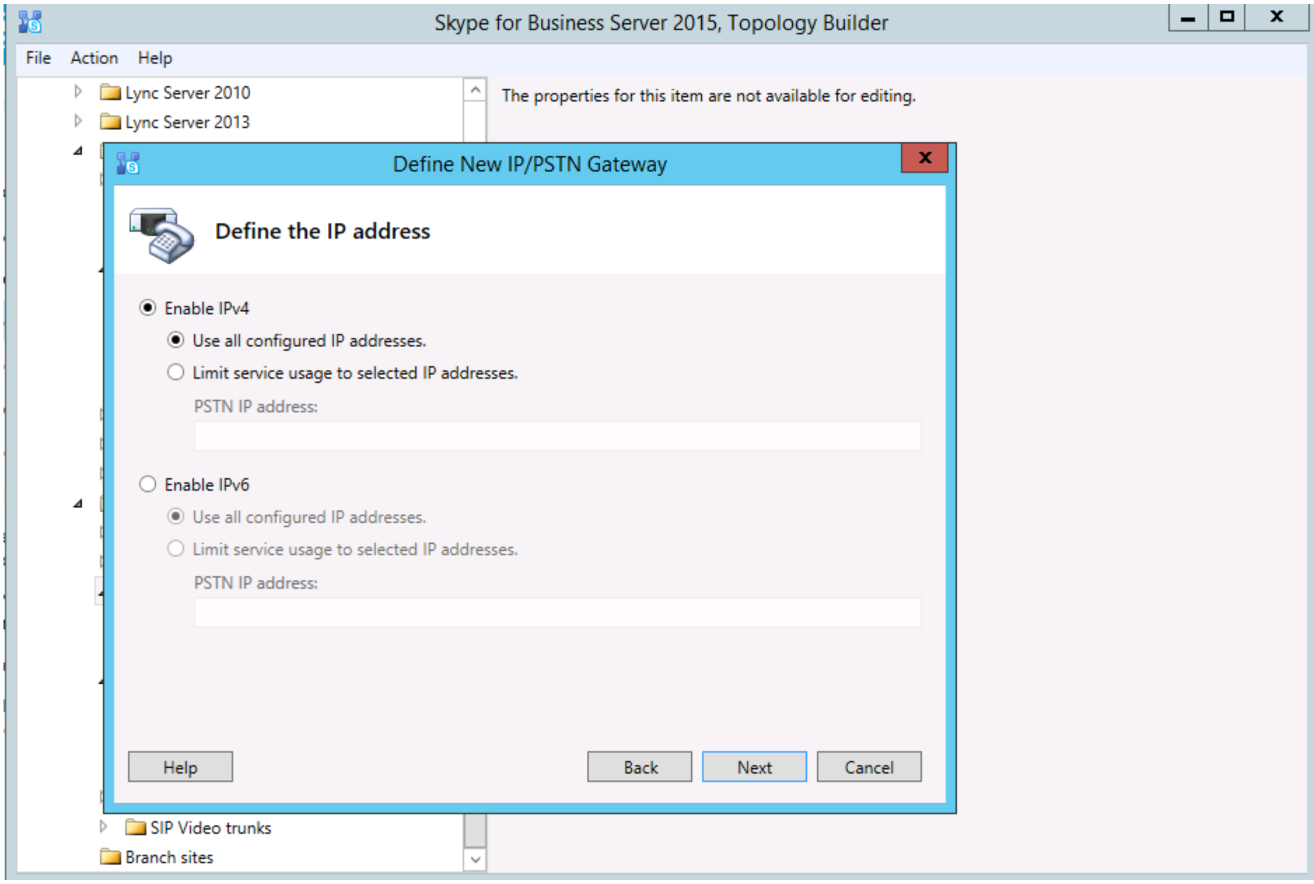
6. In the upper left hand corner, expand the site in which the PSTN gateway will be added. In our case, the site is labeled **CleanDefaultTopology**. Expand Shared Components. Then click on the **PSTN Gateways**. Right click on PSTN gateways and select **New IP/PSTN Gateway**



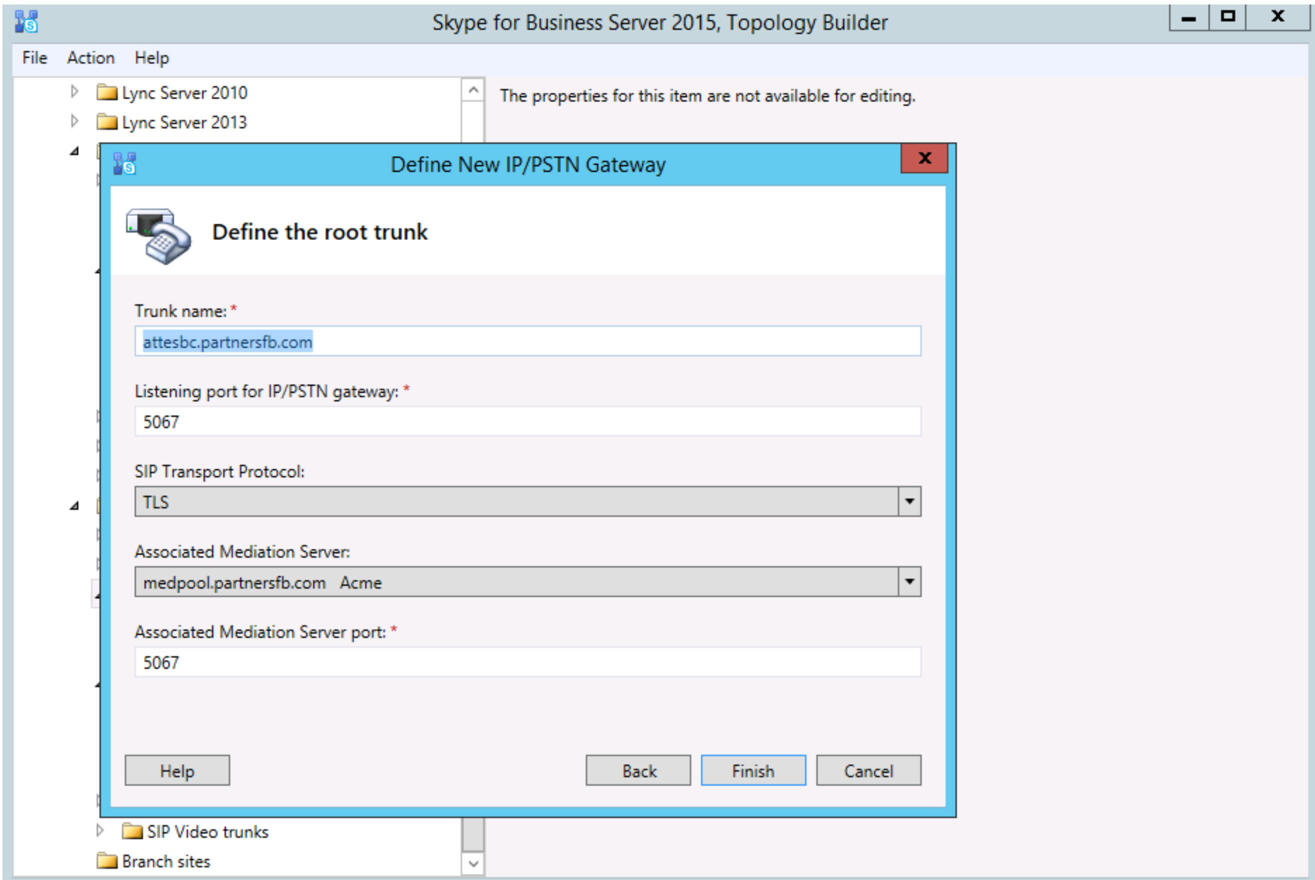
7. In the **Define New IP/PSTN Gateway** window, enter the IP address of the **SIP interface** of the ESBC in the FQDN text box and click Next.



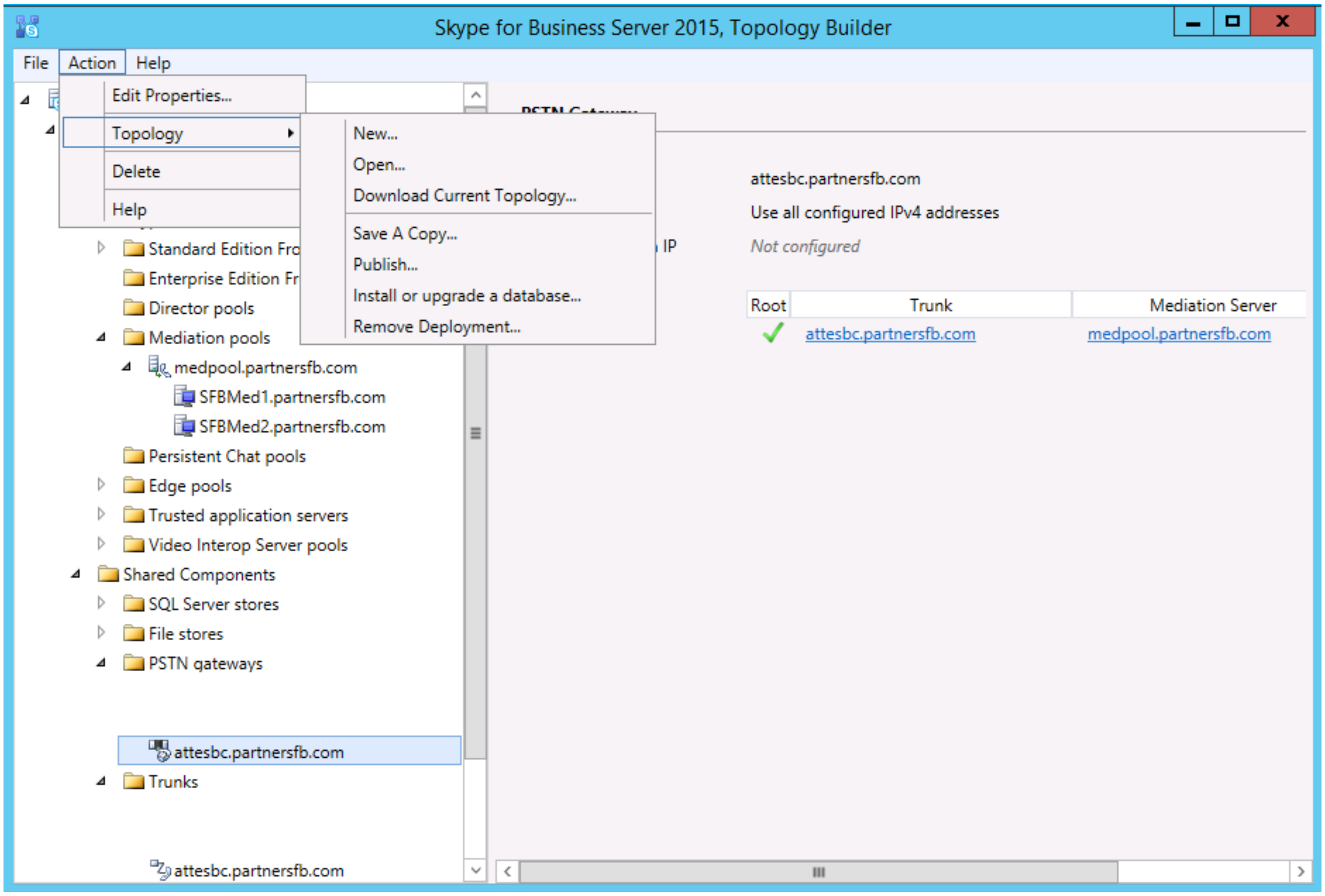
8. Select Enable IPv4 in the Define the IP address section and click Next.


Skype for Business Server 2015, Topology Builder
File
Action
Help
Lync Server 2010
Lync Server 2013
The properties for this item are not available for editing.
Define New IP/PSTN Gateway
Define the IP address
Enable IPv4
Use all configured IP addresses.
Limit service usage to selected IP addresses.
PSTN IP address:
Enable IPv6
Use all configured IP addresses.
Limit service usage to selected IP addresses.
PSTN IP address:
Help
Back
Next
Cancel
SIP Video trunks
Branch sites

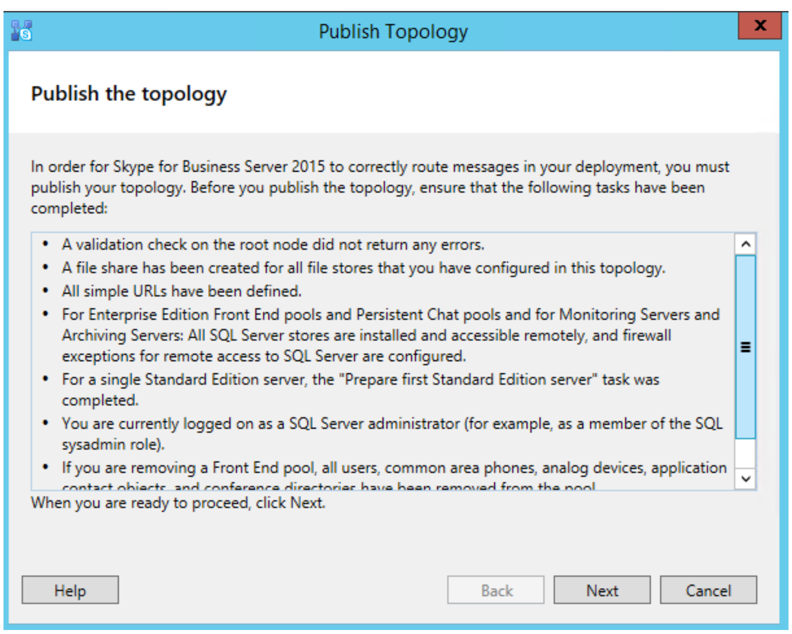
9. In the next section, enter the IP address of the ESBC's SIP interface under Trunk name. Configure the Listening port for IP/PSTN gateway as 5060, TCP as the SIP Transport Protocol, and 5060 as the Associated Mediation Server port, and click Finish.



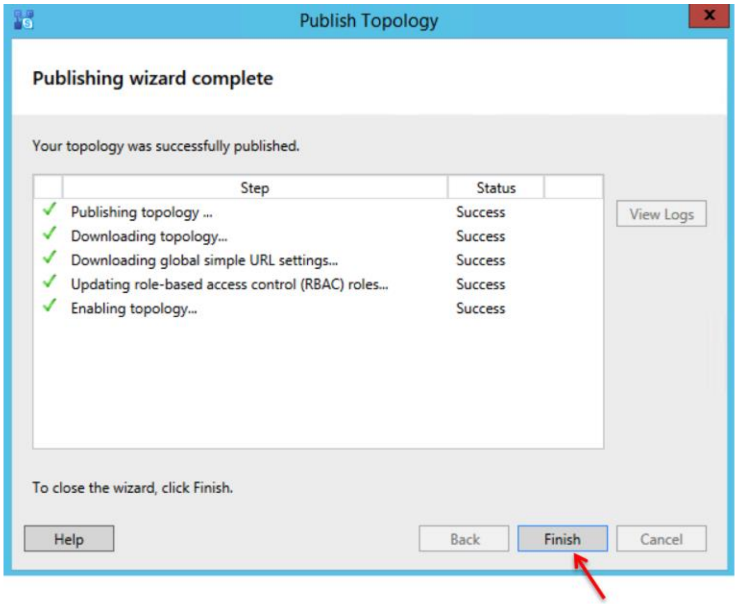
10. In the upper right hand corner of your screen under Actions select Topology then select Publish.



11. You will now see the Publish Topology window. Click on the Next button.



12. When complete you should see a window from Topology Builder stating that your topology was successfully published. Click the Finish button.



Creating a route within the Skype for Business infrastructure

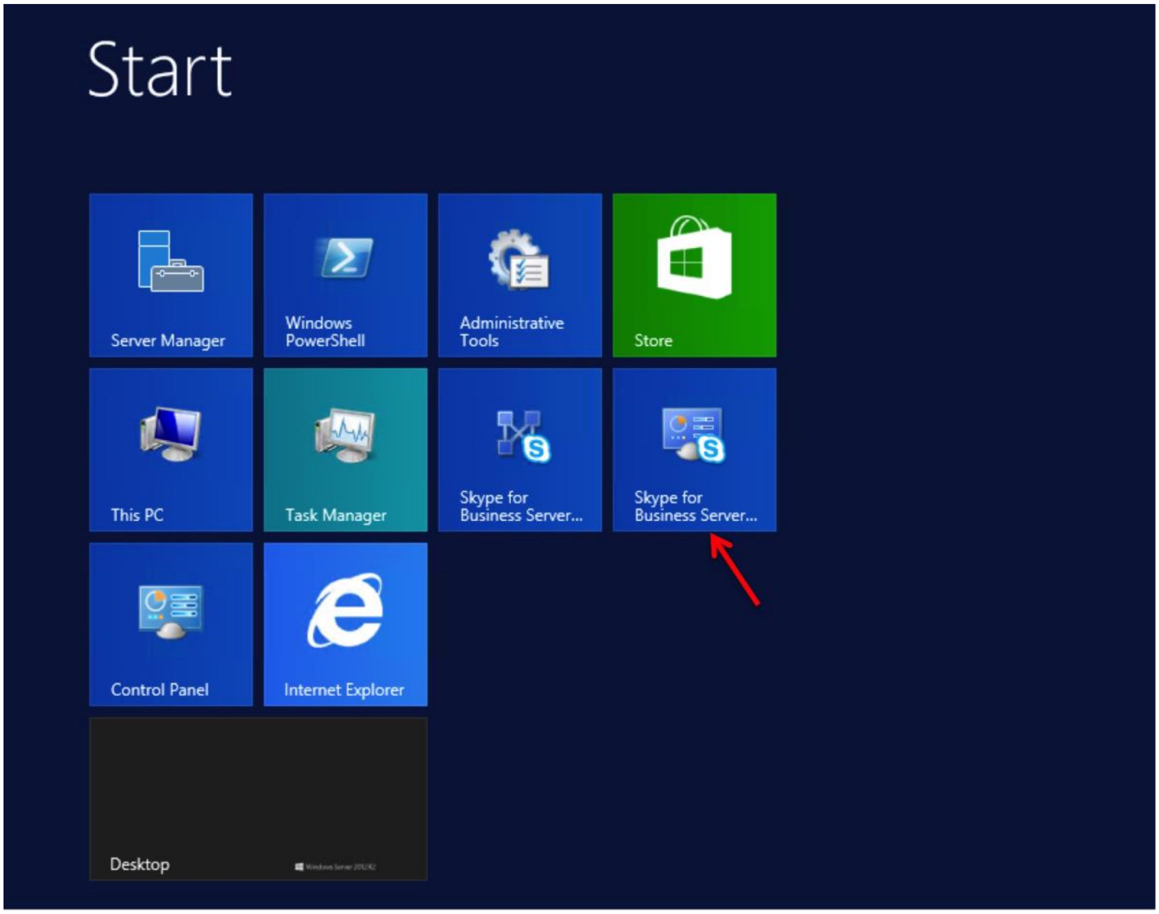
In order for the Skype for Business (SFB) clients to utilize the SIP trunking infrastructure that has been put in place, a route will need to be created to allow direction to this egress. Routes specify how SFB handles calls placed by enterprise voice users. When a user places a call, the server, if necessary, normalizes the phone number to the E.164 format and then attempts to match that phone number to a SIP Uniform Resource Identifier (URI). If the server is unable to make a match, it applies outgoing call routing logic based on the number. That logic is defined in the form of a separate voice route for each set of target phone numbers listed in the location profile for a locale. For this document we are only describing how to set up a route. Other aspects which apply to SFB deployments such as dial plans, voice policies, and PSTN usages are not covered.

To add the route we will need:

- Rights to administer the SFB Control Panel
    - Membership in the CS Administrator Active Directory Group
- Access to the SFB Control Panel

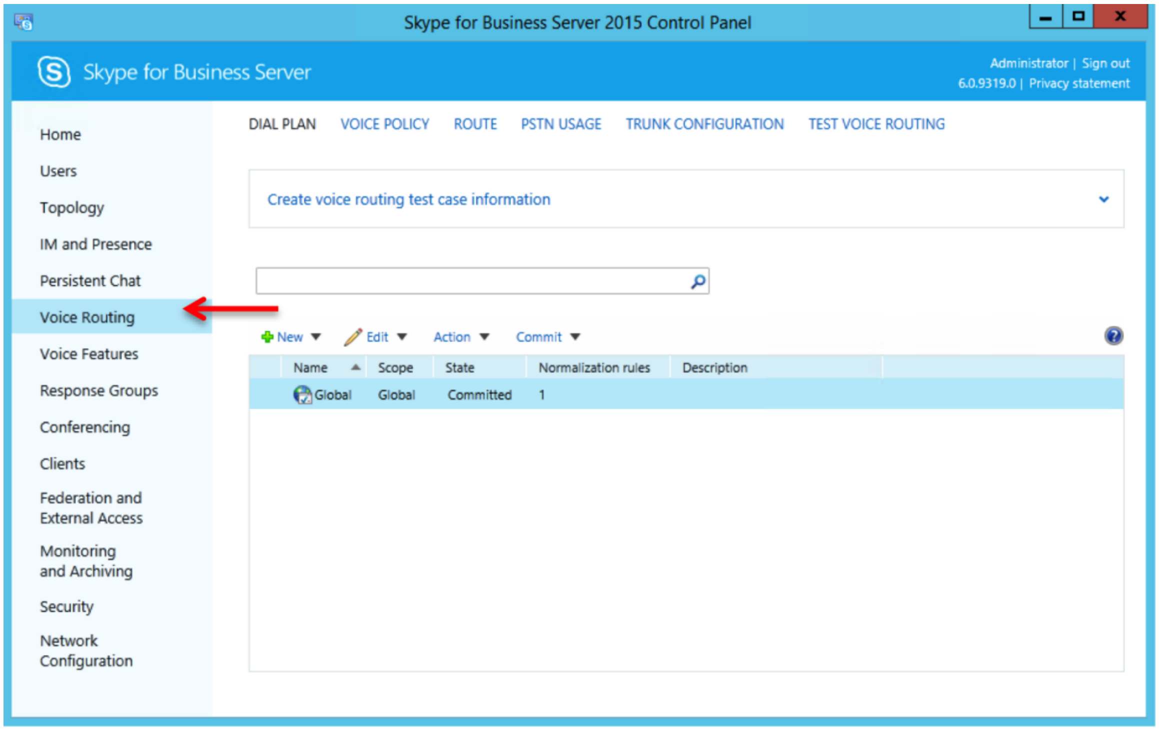
The following process details the steps to create the route:

1. From the Start bar, select SFB Control Panel.

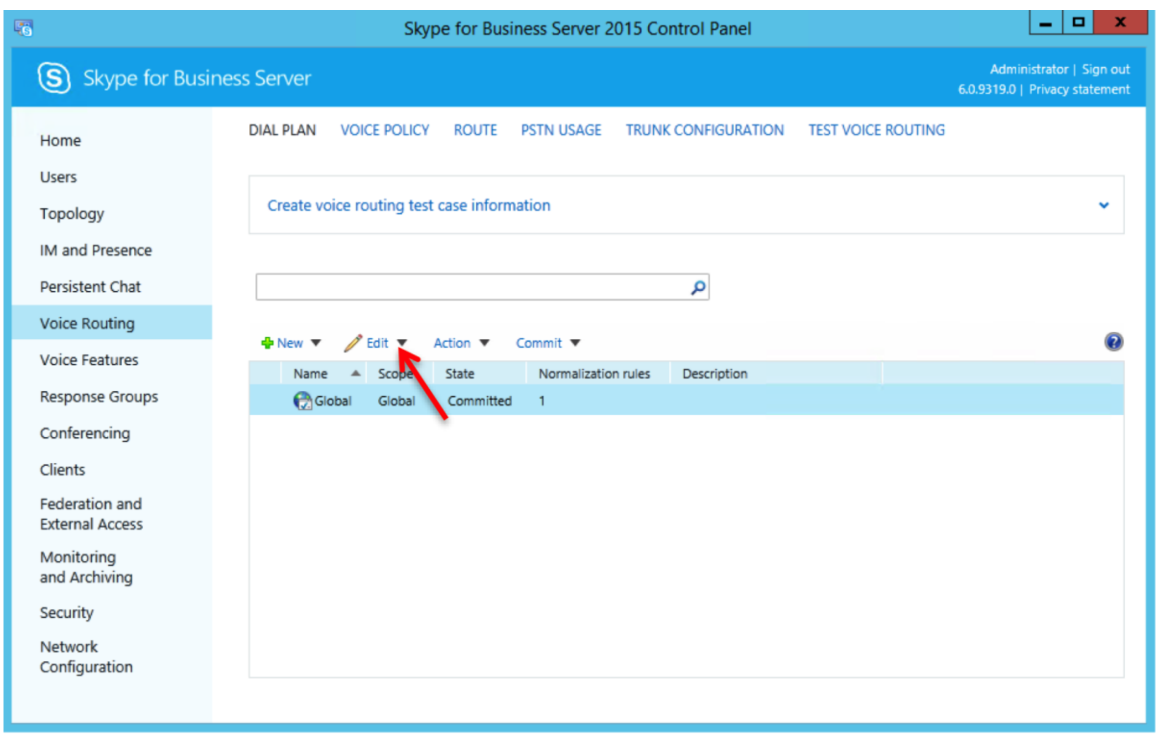


You will be prompted for credentials, enter your domain username and password.

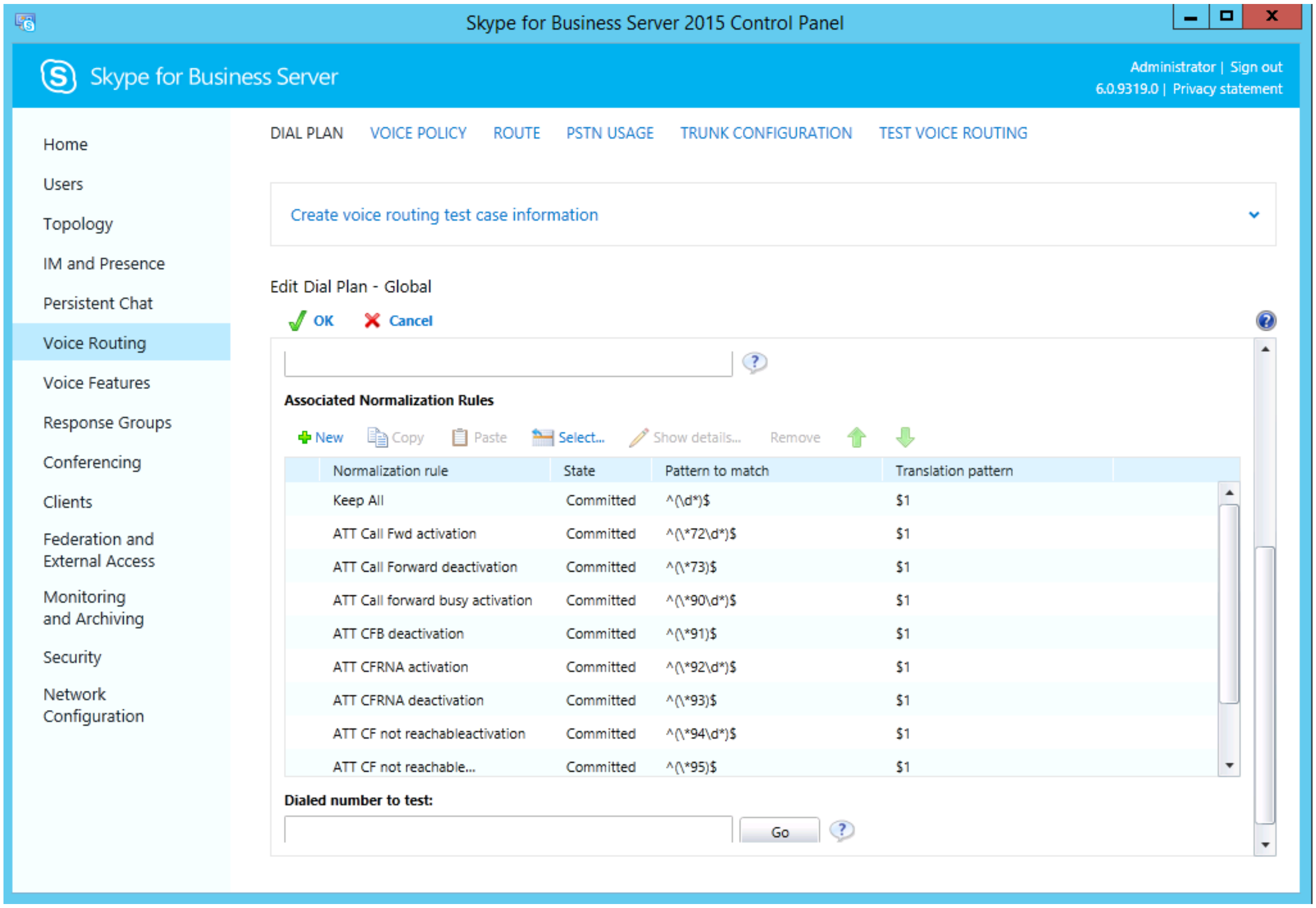
2. Once logged in, you will now be at the "Welcome Screen". On the left hand side of the window, click on Voice Routing.



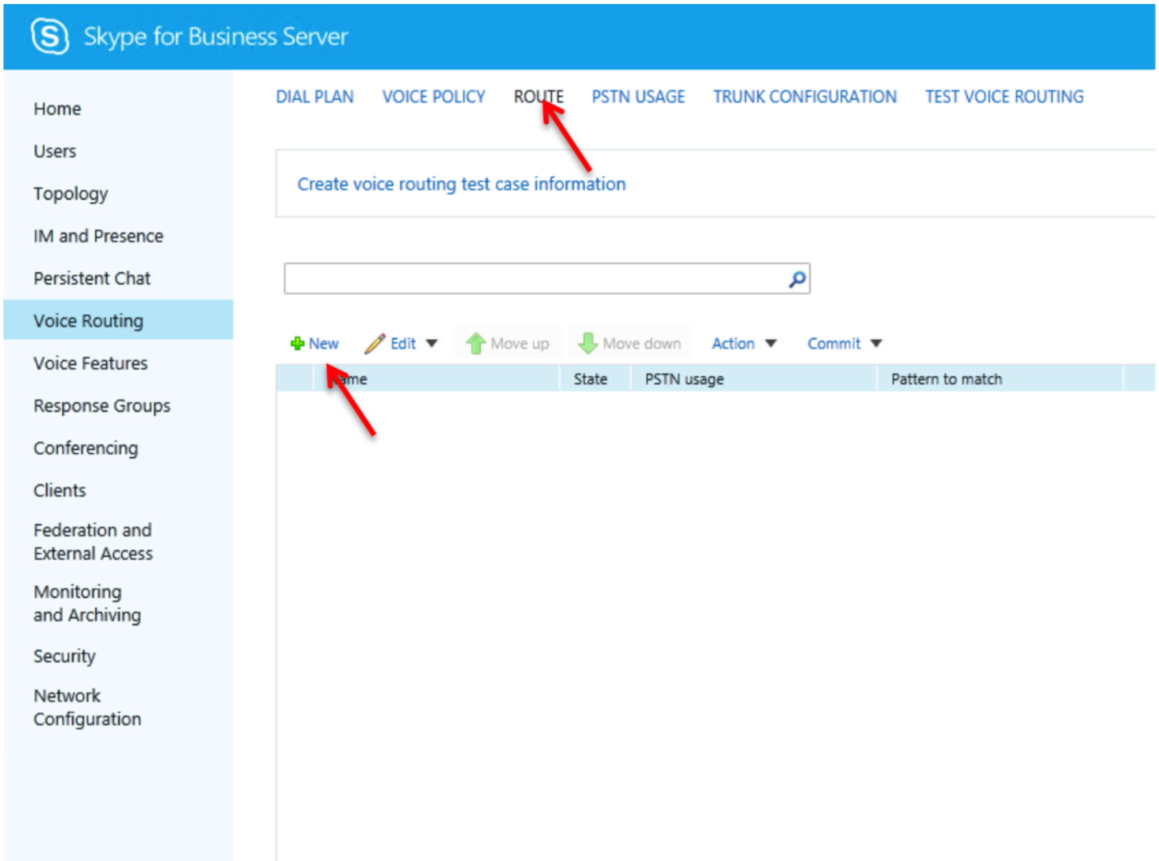
3. The Dial Plan tab in the Voice Routing section will be displayed. Select the Global dial plan. On the content area toolbar, click Edit



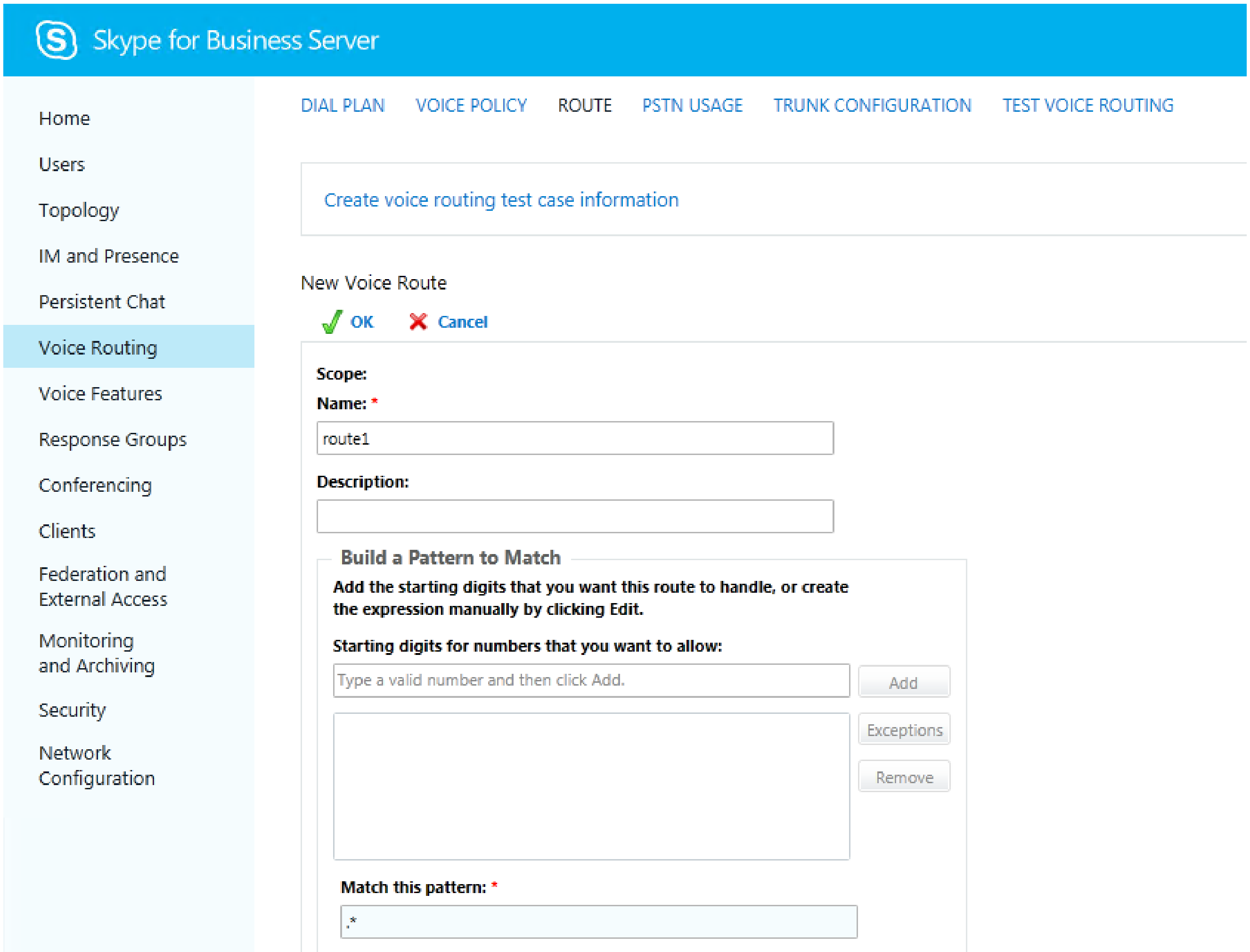
4. Next you build a Dial Plan and a translation rule for the phone numbers you want this route to handle.



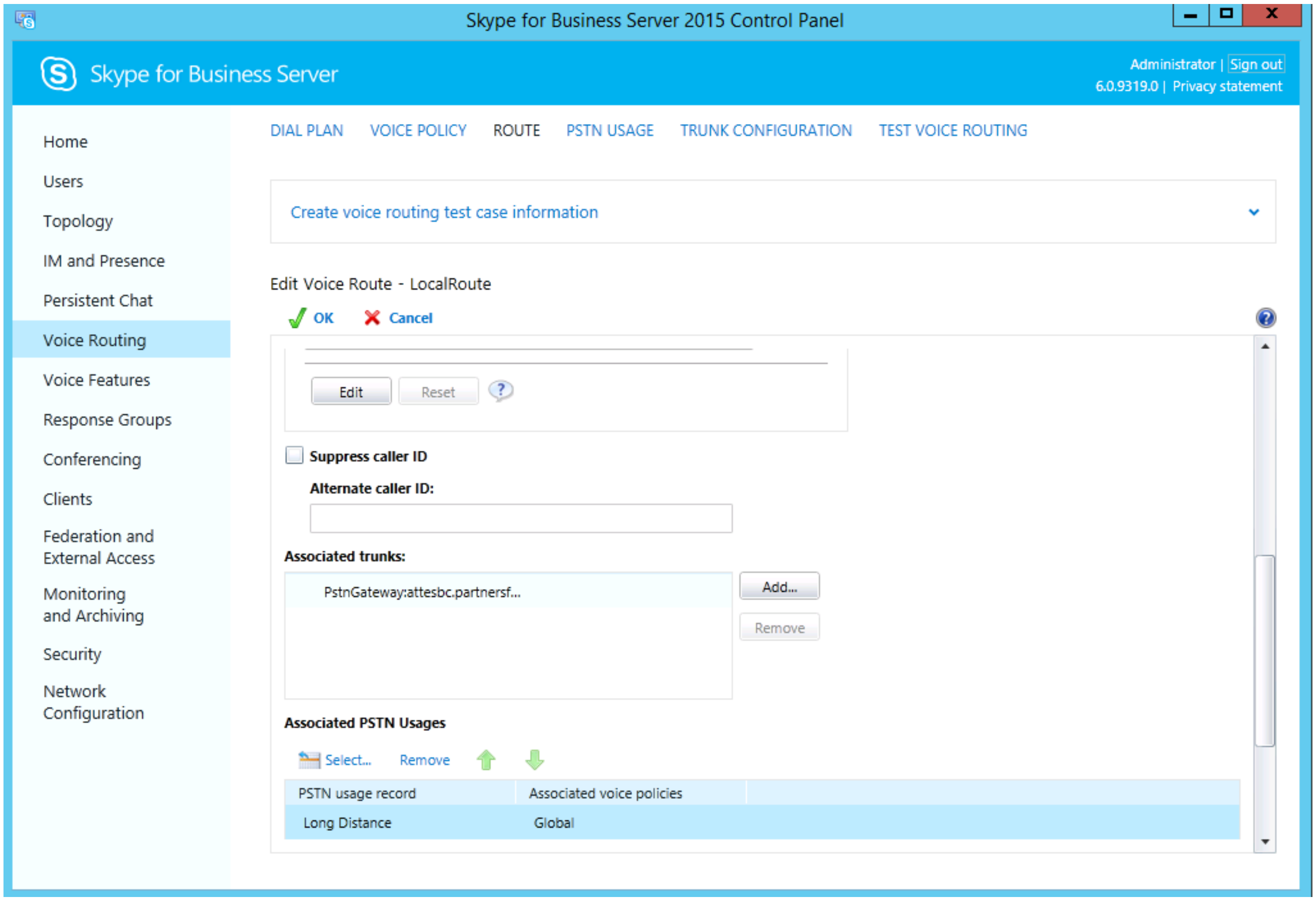
5. On the top row of the tabs, select Route. On the content area toolbar, click +New.



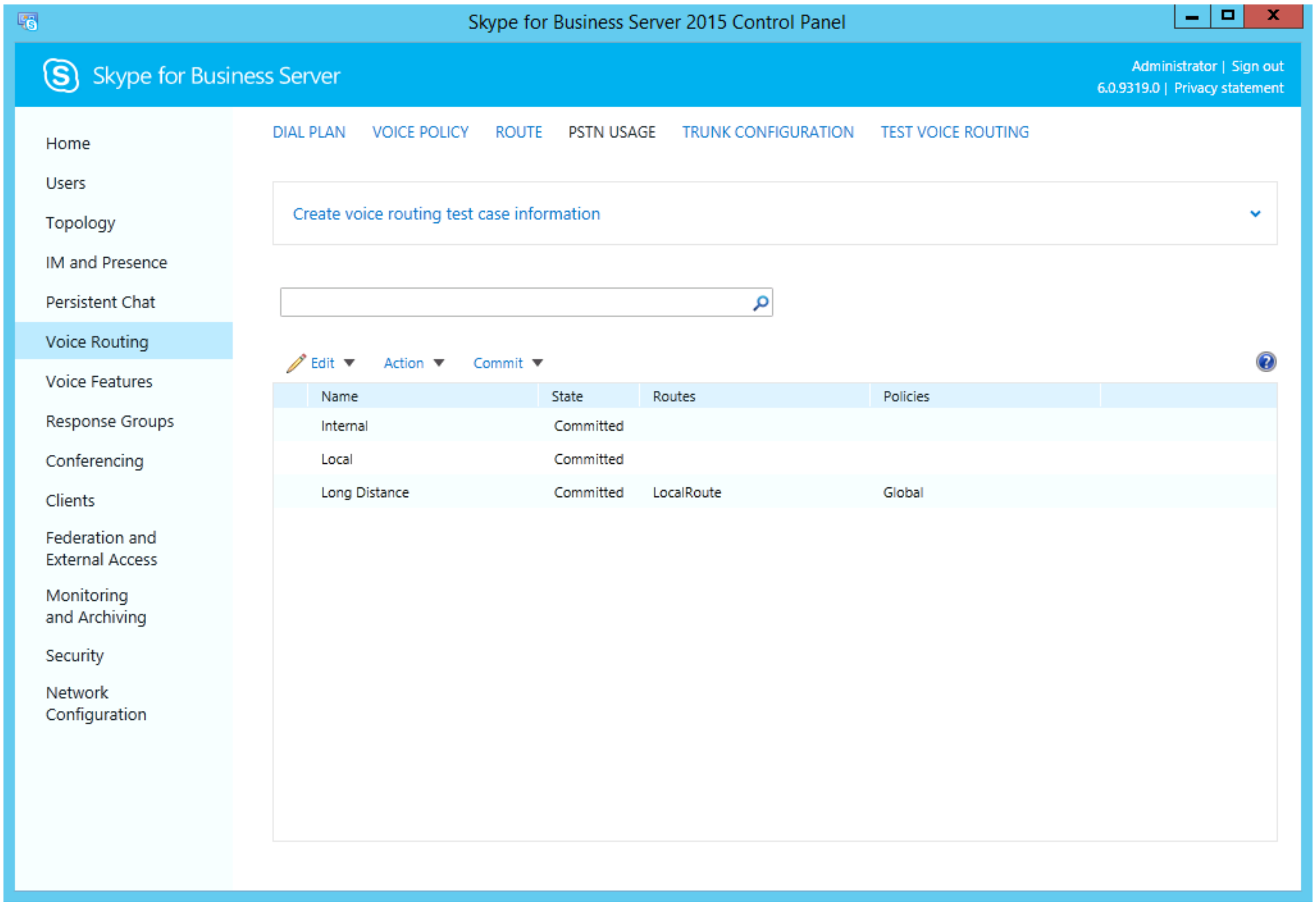
6. On the New Voice Route page, in the Name field, enter the name you have selected for the Route. In our example, it is labeled “route1”. Leave the Match this pattern field as .* so all numbers will be matched.



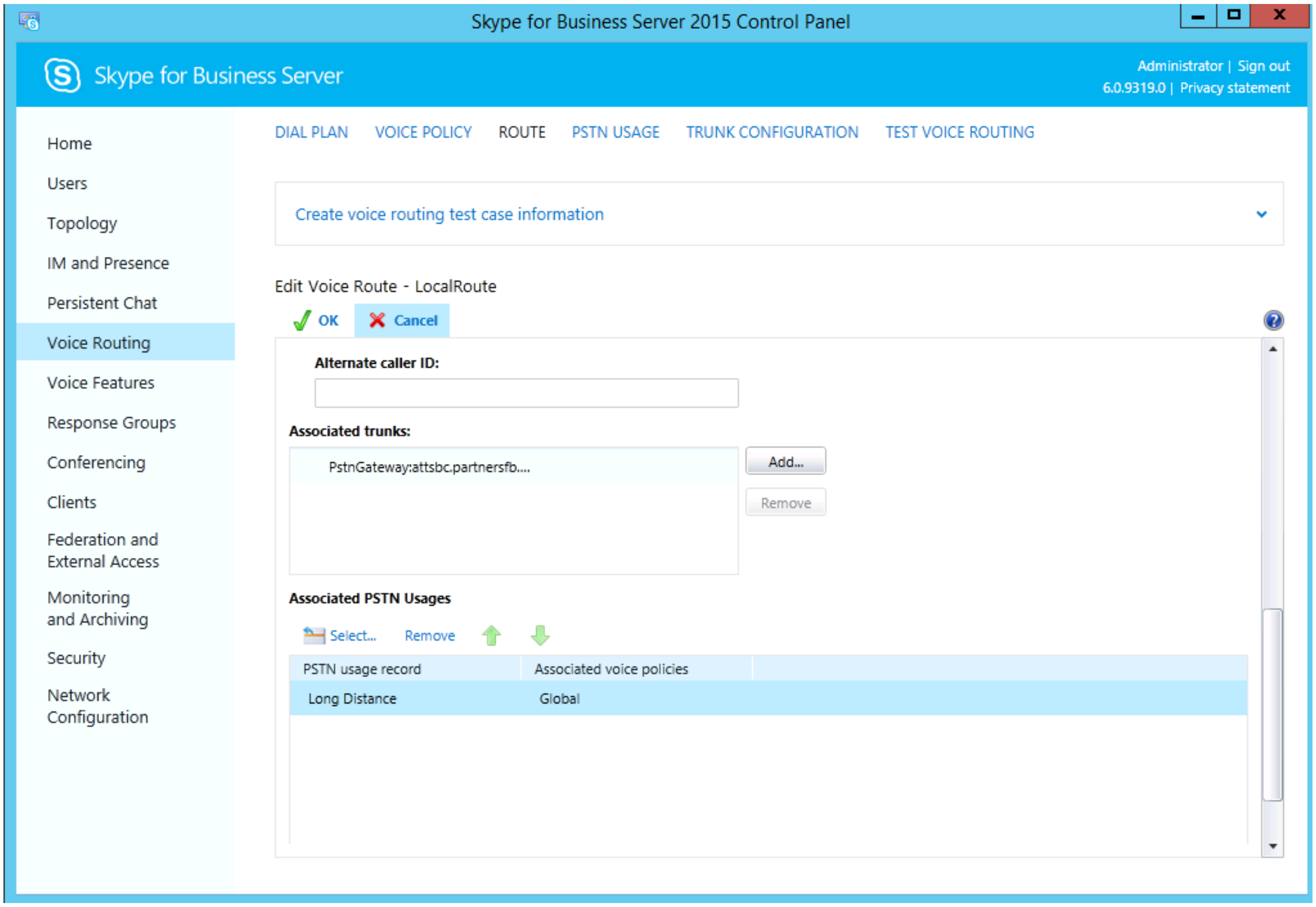
7. Next you want to associate the Voice Route with the Trunk you have just created. Scroll down to Associated Trunks and add the ATT trunk. You can now see that you have associated your trunk with the route you created. An appropriate PSTN usage record will need to be assigned as well. In our example, we use one that was already created in the enterprise. Click on the Select button under Associated PSTN Usages



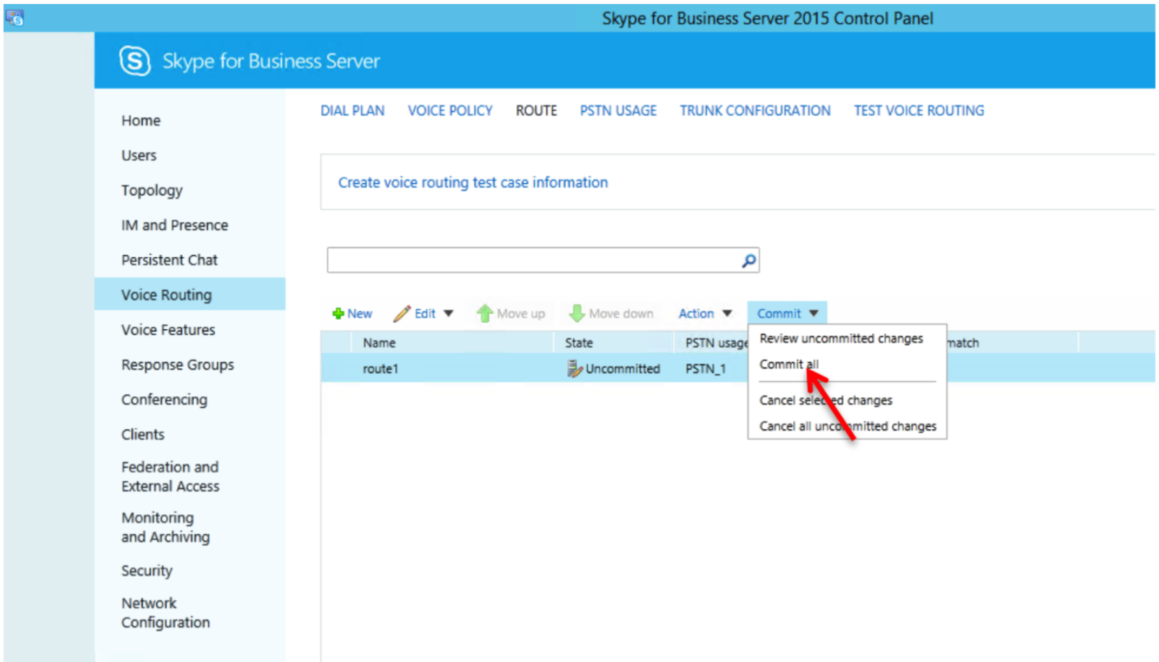
8. In the Select PSTN Usage Record window displayed, select the appropriate PSTN Usage Record and click OK.



9. You will now see the Associated PSTN Usages which you have added. Click the OK button at the top of the New Voice Route screen.



10. You will now be at the Routes page showing route1. Click the Commit drop-down menu, and then Commit All.



Phase 3 – Configuring the Oracle Enterprise Operations Monitor

In this section we describe the steps for configuring Oracle Enterprise Operations Monitor (EOM) for use with the Oracle Enterprise SBCs to monitor SIP signaling traffic on the network.

In Scope

The following guide for configuring the Oracle EOM assumes that this is a newly deployed device dedicated to a single customer. Please see the Oracle Communications Session Monitor Installation Guide on http://docs.oracle.com/cd/E60864_01/index.htm for a better understanding of the basic installation.

Out of Scope

- Basic installation as this is covered in Chapters 2 and 3 of the Oracle Communications Session Monitor Installation Guide.
- High availability.

What will you need

- Console access to the EOM server or virtual machine (VM).
- Browser-based HTTPS access to the EOM server after the initial configuration is complete.
- Administrator password for the EOM to be used.
- IP address to be assigned to EOM.

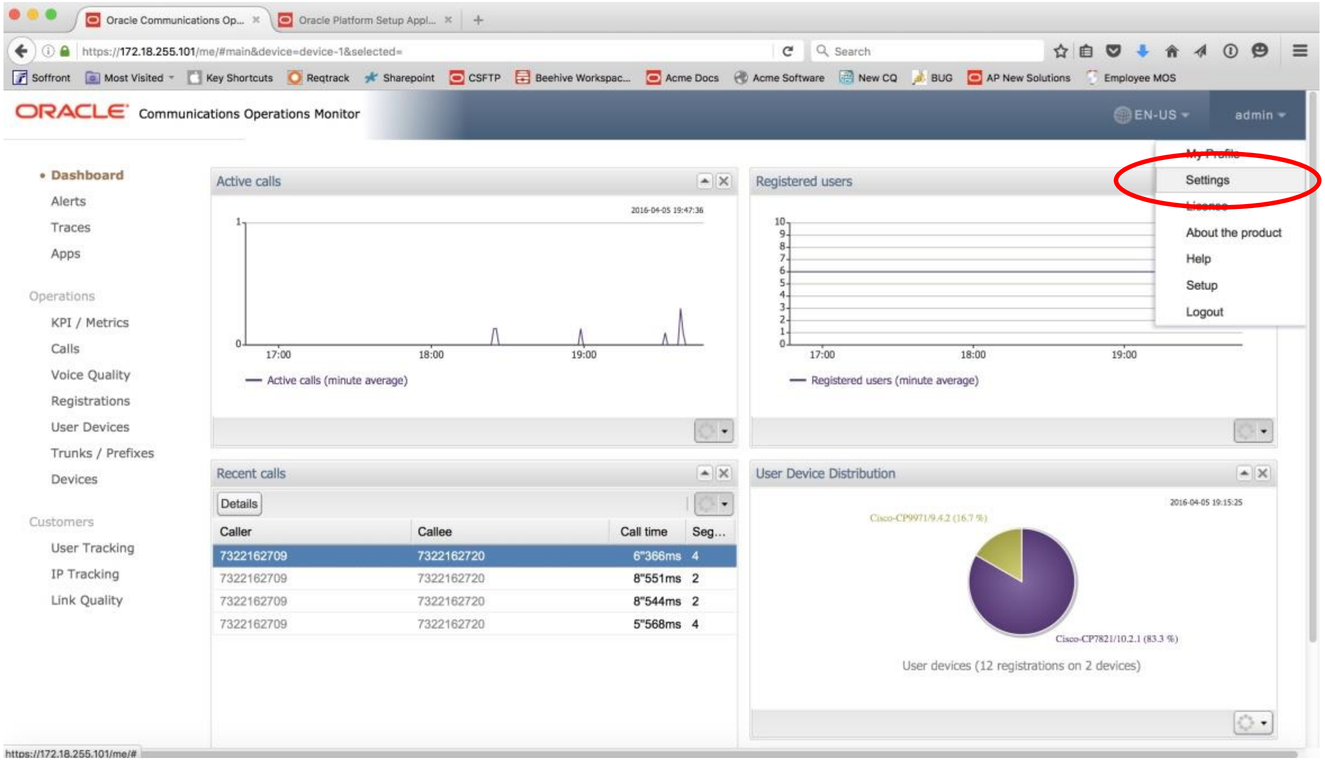
EOM – Getting Started

Ensure that the server or VM specifications meet those outlined in Chapter 1 of the Oracle Communications Session Monitor Installation Guide. Install the EOM software and configure the network parameters as outlined in Chapter 2 of the same guide. Chapter 3 details the subsequent browser-based installation. When prompted to select the “Machine Type”, select the “Communications Operations Monitor” checkbox.

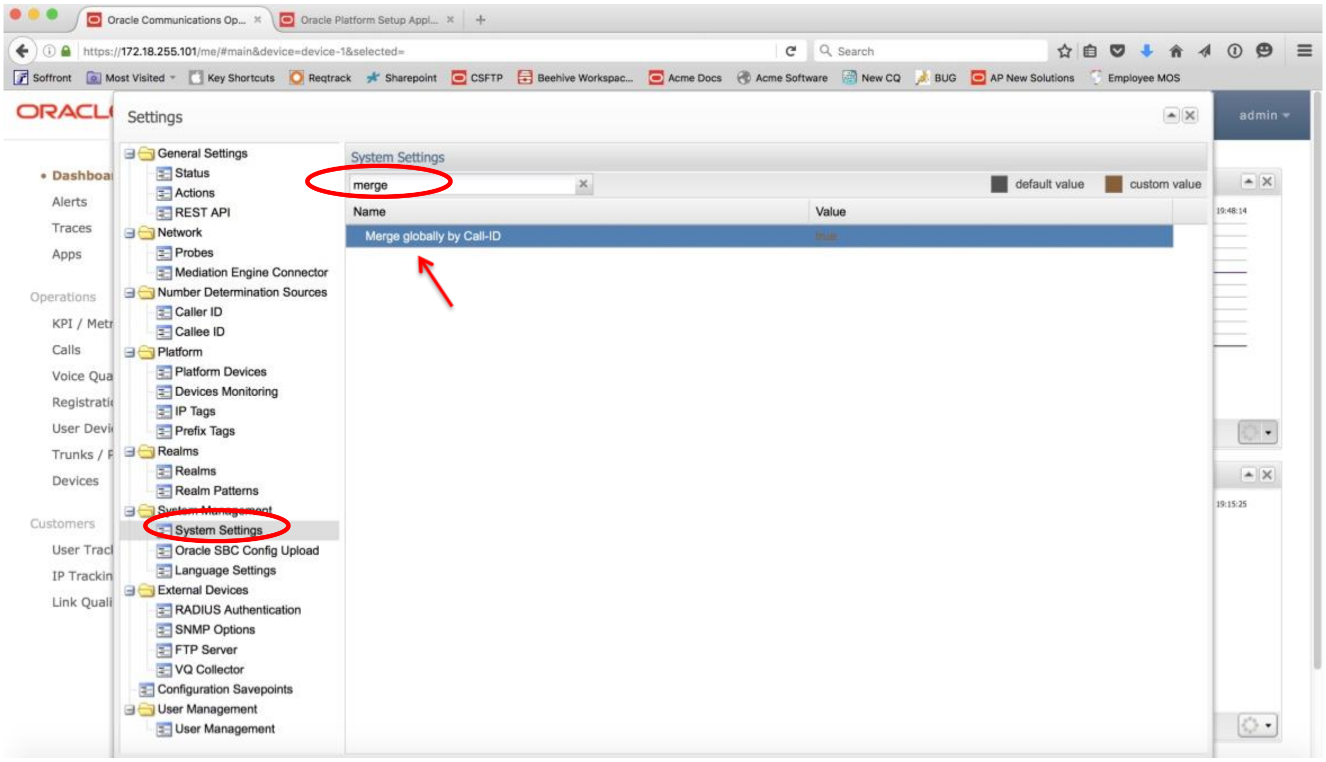
Configuring EOM to Display All Legs of a Call in a Single Report

This allows all call legs on both sides of the E-SBC to be displayed in a single report, making analysis and troubleshooting easier.

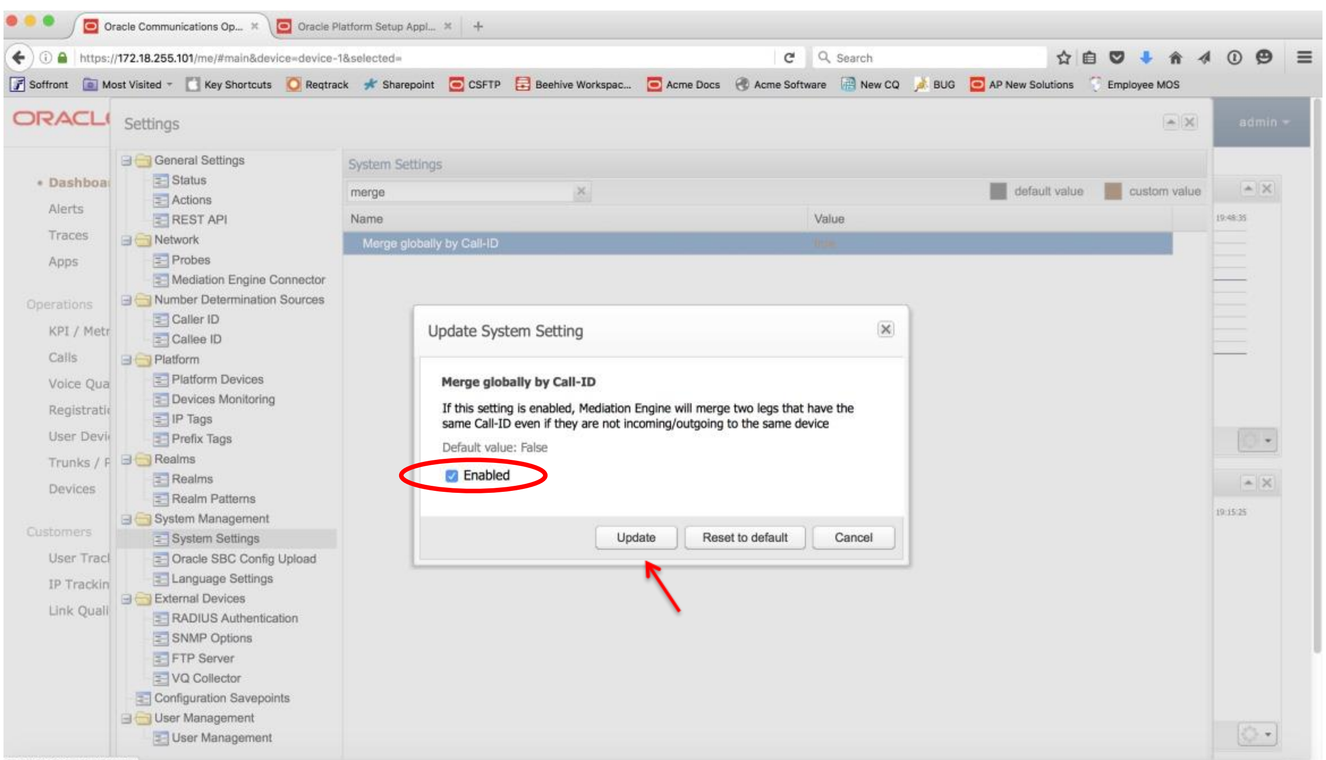
1. Click on the user (admin in this example) in the top right corner, then click on Settings.



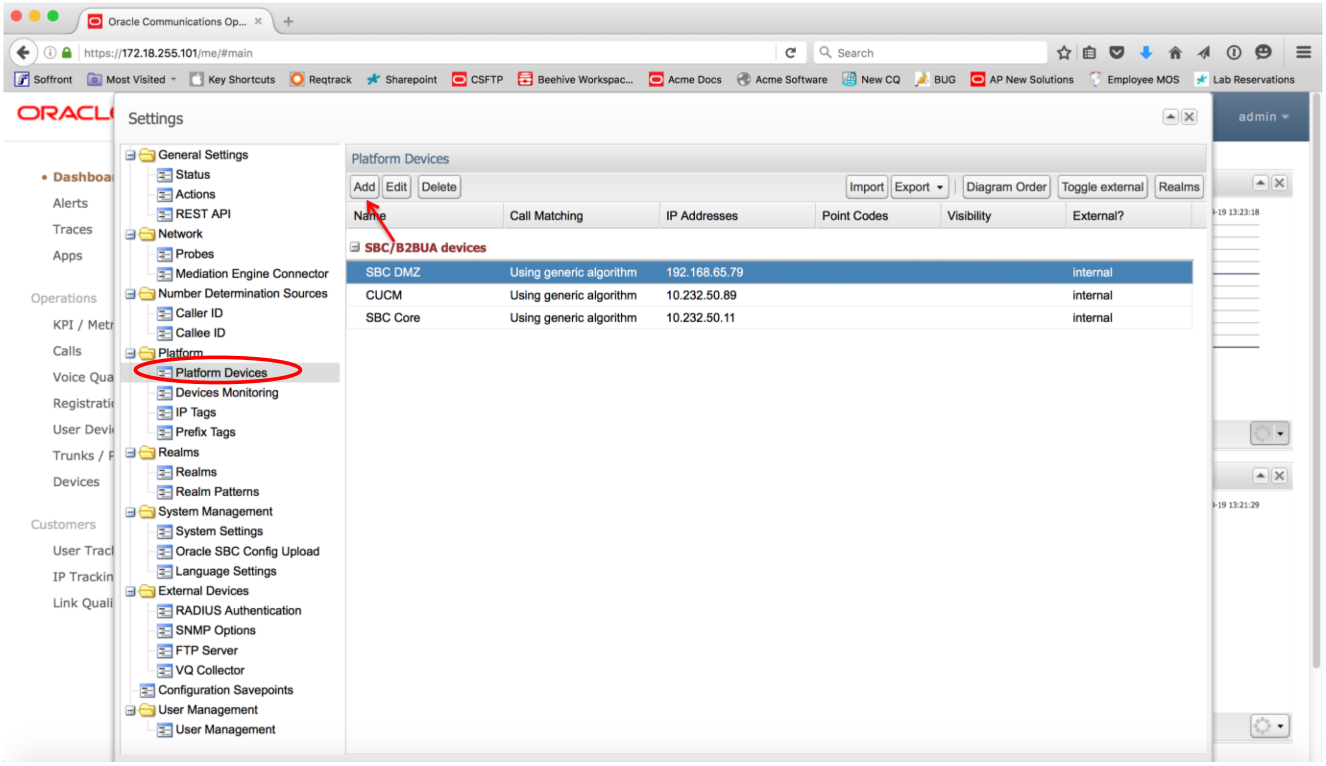
2. Under System Management select System Settings and search for “merge”. Double click on “Merge globally by Call-ID”.



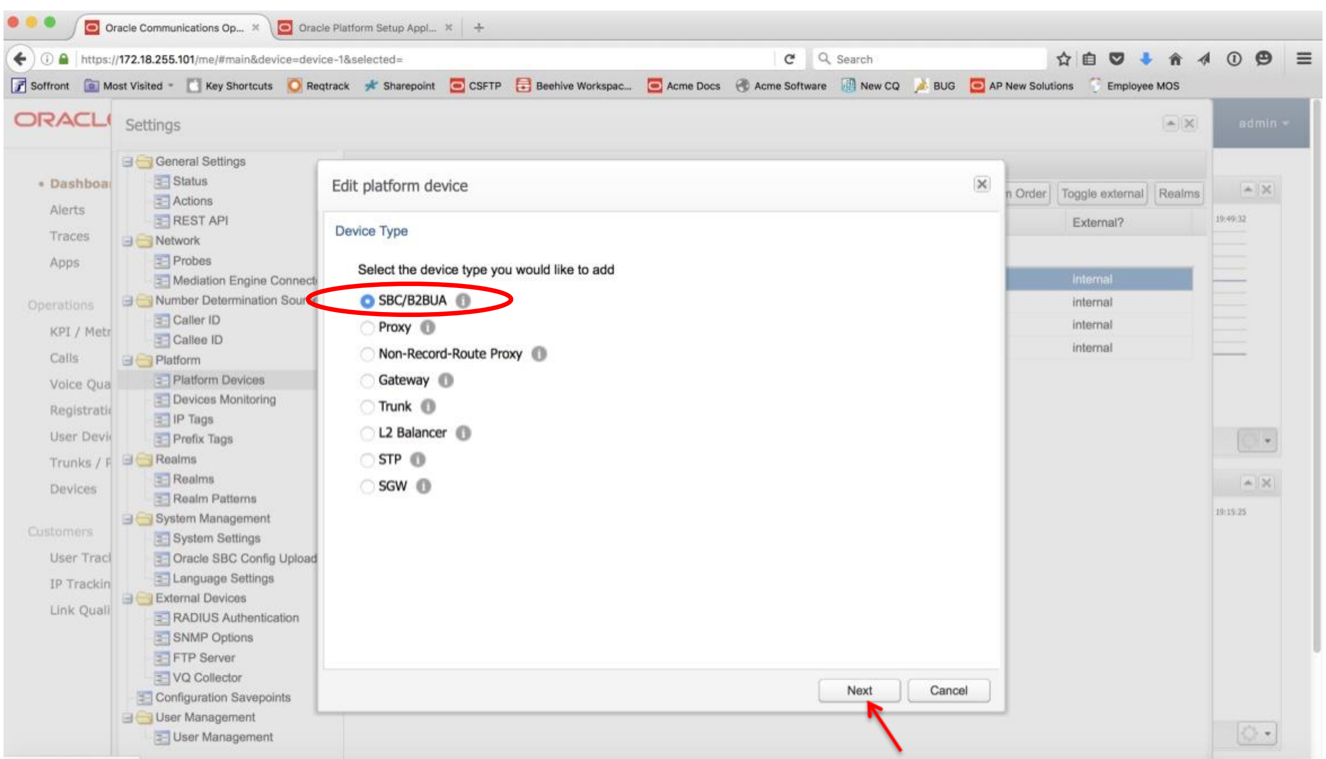
3. Click on the Enabled check box and click Update.



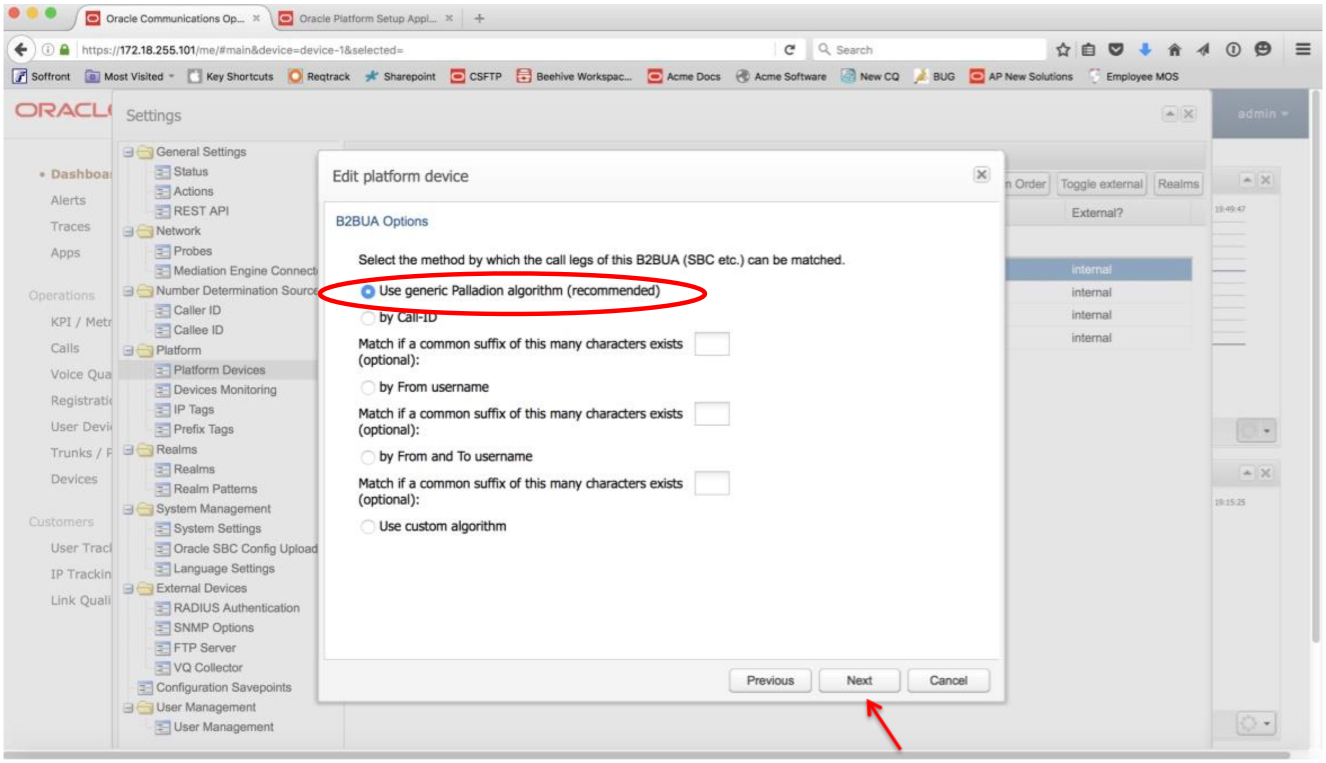
4. Under Platform select Platform Devices. Click Add (or Edit if you've already added a device).



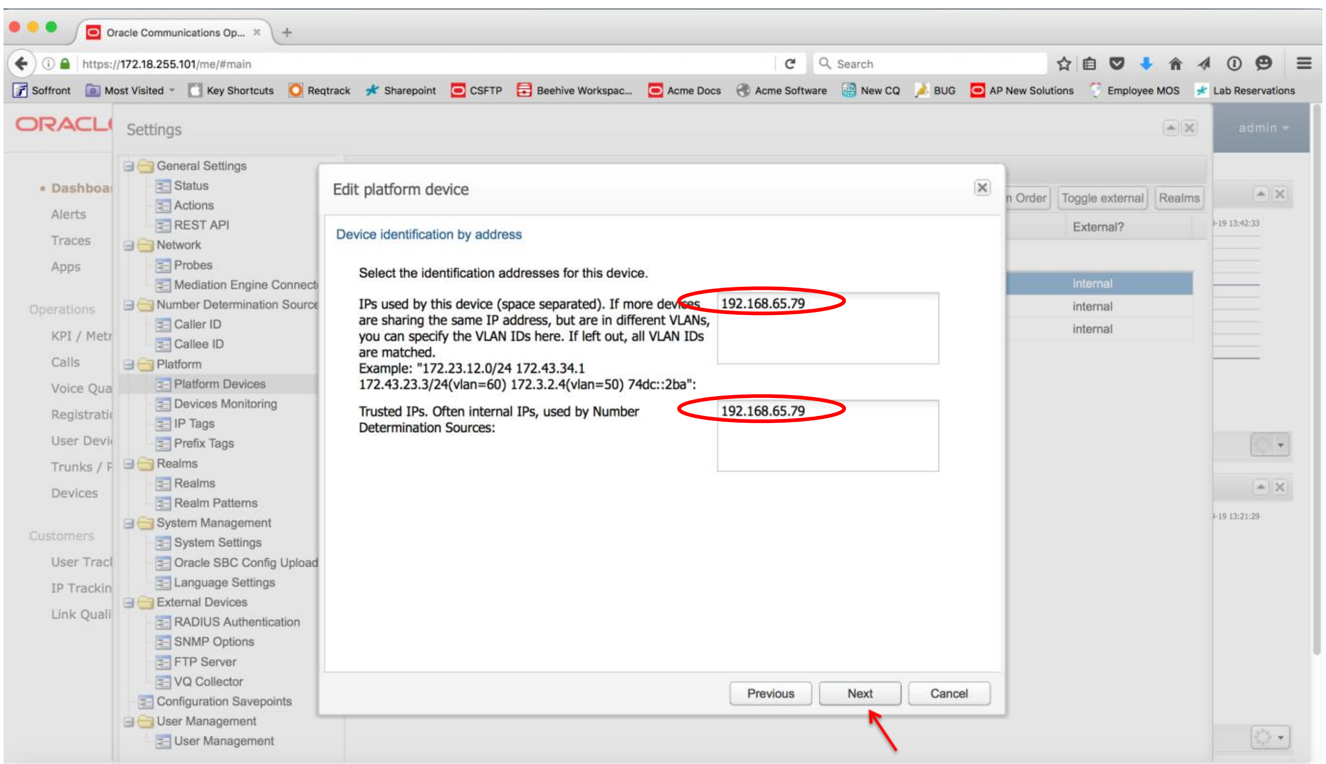
5. Select the SBC/B2BUA radio button regardless of the type of device you're adding, then click Next.



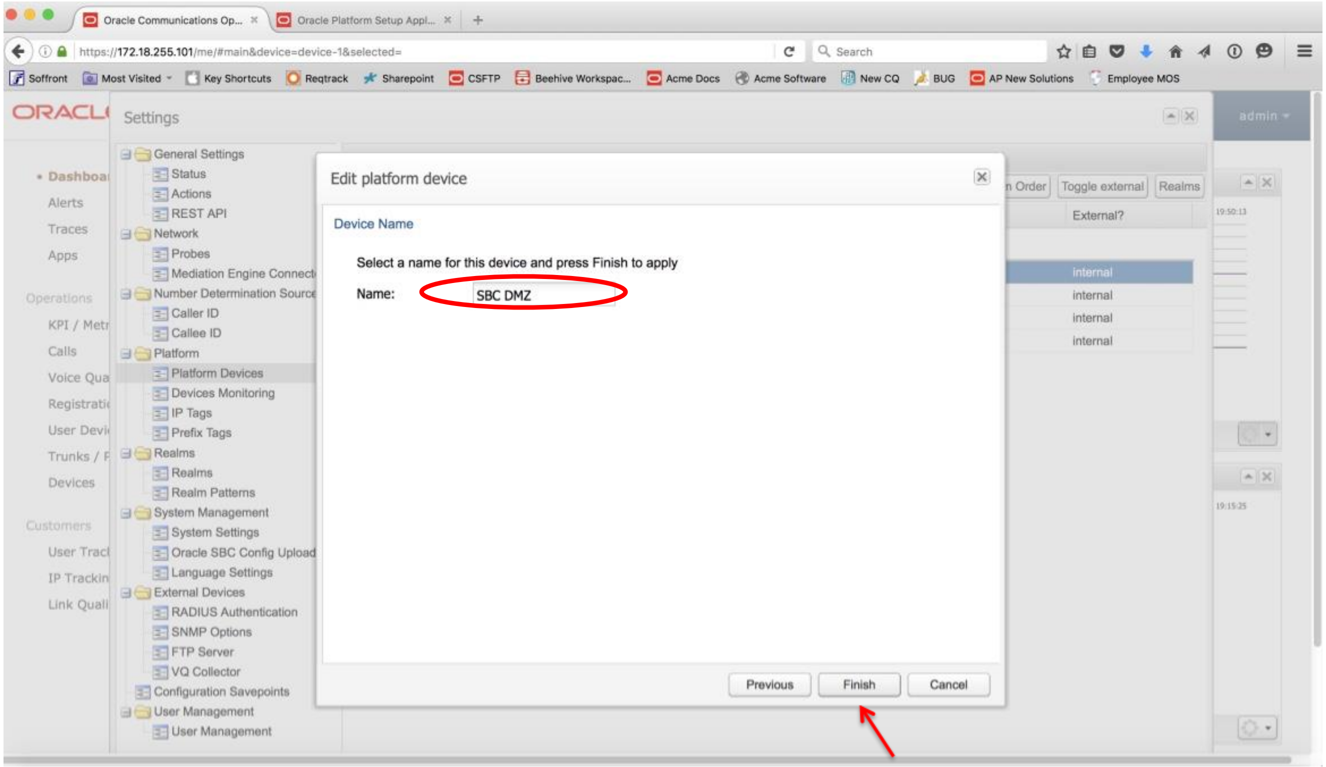
6. Click on the “Use generic Palladion algorithm (recommended)” radio button, then click Next.



7. Enter the device’s IP address in both fields, then click Next.



8. Enter a name for the device and click Finish.



9. Repeat for all other devices in the call flow. Enter each side of the SBC (inside and outside) separately. You don't necessarily need to define the access client's information.
10. On the Dashboard, under Recent Calls, make sure the Auto Refresh is set to something other than Off.
11. Make a call. After the call is finished, the call will show up under Recent Calls with 2 or more segments if the call only traverses the SBC once, or with 4 or more segments if the call traverses the SBC twice. Double click on the call.
12. The call will show up with all segments. Click on the PDF button to generate a report.
13. Click on the Create button.
14. Choose to either save the file or open it.
15. View the Call Report in Acrobat Reader or another program. The report will show all segments of the call.

Test Summary

A comprehensive test plan was executed per ATT test specifications and call flows. For a copy of full test report, please contact your Oracle Sales account team.

Troubleshooting Tools

If you find that you are not able to complete calls or have problems with the test cases, there are a few tools available for Windows Server, Lync/SFB Server, and the Oracle ESBC and SBC like logging and tracing which may be of assistance. In this section we will provide a list of tools which you can use to aid in troubleshooting any issues you may encounter.

Microsoft Network Monitor (NetMon)

NetMon is a network protocol analyzer which is freely downloadable from Microsoft. It can be found at www.microsoft.com/downloads. NetMon could be installed on the Lync Server mediation server, the Lync Server Standard Edition server, or Enterprise Edition front end server.

Wireshark

Wireshark is also a network protocol analyzer which is freely downloadable from www.wireshark.org. Wireshark could be installed on the Lync/SFB Server mediation server, the Lync/SFB Server Standard Edition server, or MCS Enterprise Edition front end server.

Eventviewer

There are several locations in the event viewer where you can find valuable information to aid in troubleshooting issues with your deployment.

With the requirement that there is a completely functioning Lync and/or SFB Server with Enterprise Voice deployment in place, there are only a few areas in which one would use the Event Viewer for troubleshooting:

- The Enterprise Voice client;
- The Lync/SFB Server Front End server;
- A Lync/SFB Server Standard Edition Server; and
- A Lync/SFB Server Mediation Server.

On the Oracle E-SBC

The Oracle SBC provide a rich set of statistical counters available from the CLI, as well as log file output with configurable detail. The follow sections detail enabling, adjusting and accessing those interfaces.

Resetting the statistical counters, enabling logging and restarting the log files.

At the console:

```
oraclesbc1# reset sipd
oraclesbc1# notify sipd debug
oraclesbc1#
enabled SIP Debugging
oraclesbc1# notify all rotate-logs
```

Examining the log files

Note: You will FTP to the management interface of the ESBC or SBC with the username user and user mode password (the default is “acme”).

```
C:\Documents and Settings\user>ftp 192.168.5.24
Connected to 192.168.85.55.
220 oraclesbc1FTP server (VxWorks 6.4) ready.
User (192.168.85.55:(none)): user
331 Password required for user.
Password: acme
230 User user logged in.
ftp> cd /ramdrv/logs
250 CWD command successful.
ftp> get sipmsg.log
200 PORT command successful.
150 Opening ASCII mode data connection for '/ramdrv/logs/sipmsg.log' (3353
bytes).
226 Transfer complete.
ftp: 3447 bytes received in 0.00Seconds 3447000.00Kbytes/sec.
ftp> get log.sipd
200 PORT command successful.
150 Opening ASCII mode data connection for '/ramdrv/logs/log.sipd' (204681
bytes).
226 Transfer complete.
ftp: 206823 bytes received in 0.11Seconds 1897.46Kbytes/sec.
ftp> bye
221 Goodbye.
```

You may now examine the log files with the text editor of your choice.

Through the Web GUI

You can also check the display results of filtered SIP session data from the Oracle E-SBC and ESBC, and provide traces in a common log format for local viewing or for exporting to your PC. Please check the "Monitor and Trace SIP Messages" section (page 140) of the E-SBC Web GUI User Guide available at http://docs.oracle.com/cd/E56581_01/index.htm. For the ESBC, see the "Monitor and Trace" section (page 95) of the User's Guide available at http://docs.oracle.com/cd/E55725_01/index.htm.

Telnet

Since we are working within an architecture which uses bound TCP listening ports for functionality, the simplest form of troubleshooting can be seeing if the devices are listening on a particular port, as well as confirming that the there is nothing blocking them such as firewalls. Ensure that you have a TELNET client available on a workstation.

All devices tested in this document will listen on TCP port 5060 for SIP signaling. In our example we are listening on 5060 on the PSTN facing NIC. Tests may include:

- Client to pool server: telnet <servername> 5060
- Pool server to Mediation Server: telnet <servername> 5060

Appendix A

Accessing the ACLI

Access to the ACLI is provided by:

- The serial console connection;
- TELNET, which is enabled by default but may be disabled; and
- SSH.

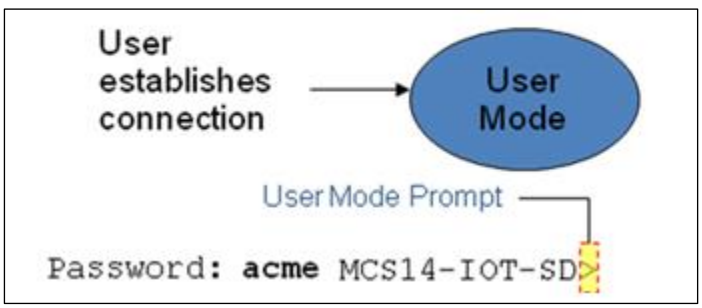
Initial connectivity will be through the serial console port. At a minimum, this is how to configure the management (eth0) interface on the SBC.

ACLI Basics

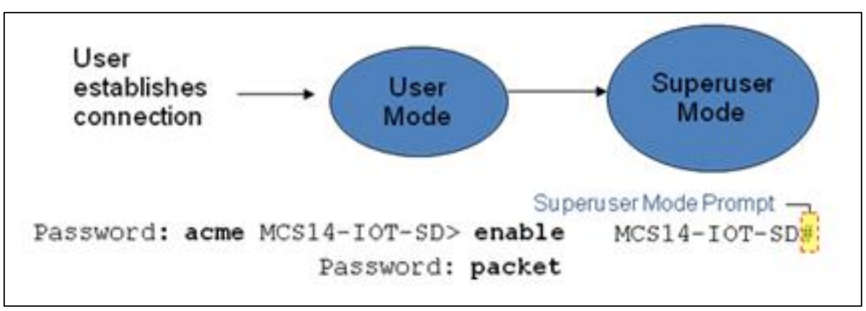
There are two password protected modes of operation within the ACLI, User mode and Superuser mode.

When you establish a connection to the SBC, the prompt for the User mode password appears. The default password is acme.

User mode consists of a restricted set of basic monitoring commands and is identified by the greater than sign (>) in the system prompt after the target name. You cannot perform configuration and maintenance from this mode.



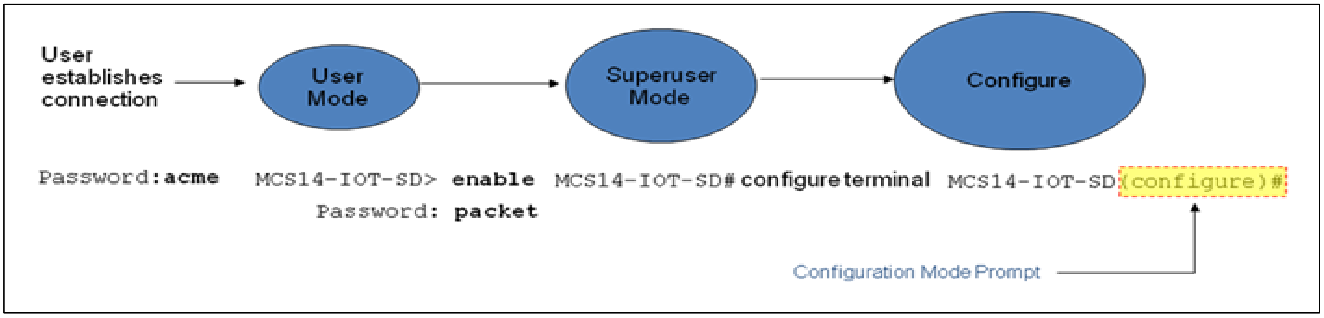
The Superuser mode allows for access to all system commands for operation, maintenance, and administration. This mode is identified by the pound sign (#) in the prompt after the target name. To enter the Superuser mode, issue the enable command in the User mode.



From the Superuser mode, you can perform monitoring and administrative tasks; however you cannot configure any elements. To return to User mode, issue the exit command.

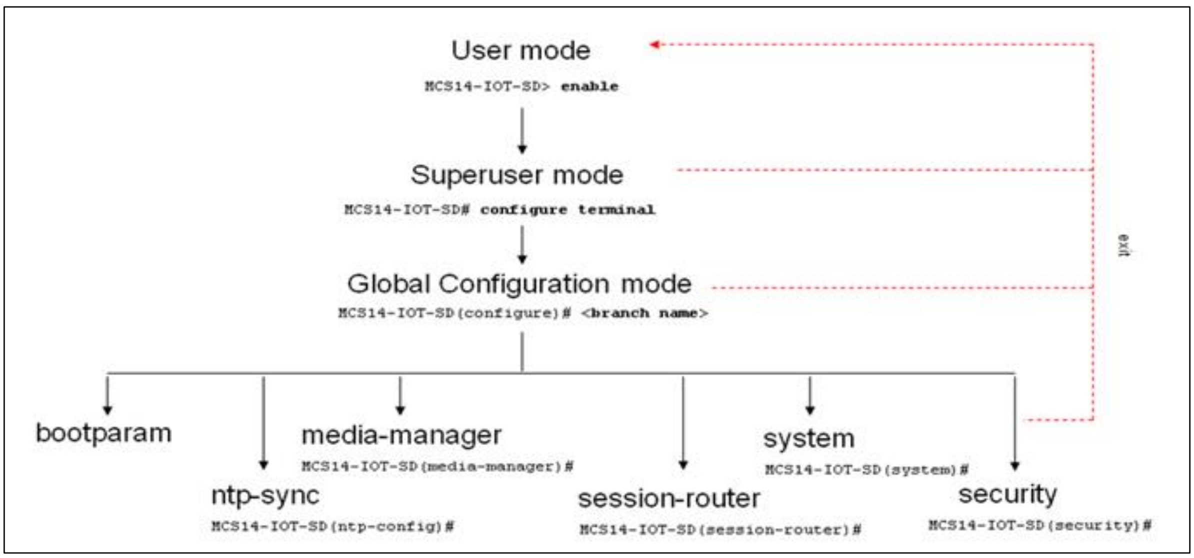
You must enter the Configuration mode to configure elements. For example, you can access the configuration branches and configuration elements for signaling and media configurations. To enter the Configuration mode, issue the configure terminal command in the Superuser mode.

Configuration mode is identified by the word configure in parenthesis followed by the pound sign (#) in the prompt after the target name, for example, oraclesbc1(configure)#. To return to the Superuser mode, issue the exit command.



In the configuration mode, there are six configuration branches:

- bootparam;
- ntp-sync;
- media-manager;
- session-router;
- system; and
- security.



The ntp-sync and bootparams branches are flat branches (i.e., they do not have elements inside the branches). The rest of the branches have several elements under each of the branches.

The bootparam branch provides access to SBC boot parameters.

The ntp-sync branch provides access to ntp server configuration commands for synchronizing the SBC time and date.

The security branch provides access to security configuration.

The system branch provides access to basic configuration elements as system-config, snmp-community, redundancy, physical interfaces, network interfaces, etc.

The session-router branch provides access to signaling and routing related elements, including H323-config, sip-config, iwf-config, local-policy, sip-manipulation, session-agent, etc.

The media-manager branch provides access to media-related elements, including realms, steering pools, dns-config, media-manager, and so forth.

You will use media-manager, session-router, and system branches for most of your working configuration.

Configuration Elements

The configuration branches contain the configuration elements. Each configurable object is referred to as an element. Each element consists of a number of configurable parameters.

Some elements are single-instance elements, meaning that there is only one of that type of the element - for example, the global system configuration and redundancy configuration.

Some elements are multiple-instance elements. There may be one or more of the elements of any given type. For example, physical and network interfaces.

Some elements (both single and multiple instance) have sub-elements. For example:

- SIP-ports - are children of the sip-interface element
- peers – are children of the redundancy element
- destinations – are children of the peer element

Creating an Element

1. To create a single-instance element, you go to the appropriate level in the ACLI path and enter its parameters. There is no need to specify a unique identifier property because a single-instance element is a global element and there is only one instance of this element.
2. When creating a multiple-instance element, you must specify a unique identifier for each instance of the element.
3. It is important to check the parameters of the element you are configuring before committing the changes. You do this by issuing the show command before issuing the done command. The parameters that you did not configure are filled with either default values or left empty.
4. On completion, you must issue the done command. The done command causes the configuration to be echoed to the screen and commits the changes to the volatile memory. It is a good idea to review this output to ensure that your configurations are correct.
5. Issue the exit command to exit the selected element.

Note that the configurations at this point are not permanently saved yet. If the SBC reboots, your configurations will be lost.

Editing an Element

The procedure of editing an element is similar to creating an element, except that you must select the element that you will edit before editing it.

1. Enter the element that you will edit at the correct level of the ACLI path.
2. Select the element that you will edit, and view it before editing it.
   The select command loads the element to the volatile memory for editing. The show command allows you to view the element to ensure that it is the right one that you want to edit.
3. Once you are sure that the element you selected is the right one for editing, edit the parameter one by one. The new value you provide will overwrite the old value.

4. It is important to check the properties of the element you are configuring before committing it to the volatile memory. You do this by issuing the show command before issuing the done command.
5. On completion, you must issue the done command.
6. Issue the exit command to exit the selected element.

Note that the configurations at this point are not permanently saved yet. If the SBC reboots, your configurations will be lost.

Deleting an Element

The no command deletes an element from the configuration in editing.

To delete a single-instance element,

1. Enter the no command from within the path for that specific element
2. Issue the exit command.

To delete a multiple-instance element,

1. Enter the no command from within the path for that particular element.
   The key field prompt, such as <name>:<sub-port-id>, appears.
2. Use the <Enter> key to display a list of the existing configured elements.
3. Enter the number corresponding to the element you wish to delete.
4. Issue the select command to view the list of elements to confirm that the element was removed.

Note that the configuration changes at this point are not permanently saved yet. If the SBC reboots, your configurations will be lost.

Configuration Versions

At any time, three versions of the configuration can exist on the SBC: the edited configuration, the saved configuration, and the running configuration.

- The edited configuration – this is the version that you are making changes to. This version of the configuration is stored in the SBC's volatile memory and will be lost on a reboot.
  To view the editing configuration, issue the show configuration command.
- The saved configuration – on issuing the save-config command, the edited configuration is copied into the non-volatile memory on the SBC and becomes the saved configuration. Because the saved configuration has not been activated yet, the changes in the configuration will not take effect. On reboot, the last activated configuration (i.e., the last running configuration) will be loaded, not the saved configuration.
- The running configuration is the saved then activated configuration. On issuing the activate-config command, the saved configuration is copied from the non-volatile memory to the volatile memory. The saved configuration is activated and becomes the running configuration. Although most of the configurations can take effect once being activated without reboot, some configurations require a reboot for the changes to take effect.
  To view the running configuration, issue command show running-config.

Saving the Configuration

The save-config command stores the edited configuration persistently.

Because the saved configuration has not been activated yet, changes in configuration will not take effect. On reboot, the last activated configuration (i.e., the last running configuration) will be loaded. At this stage, the saved configuration is different from the running configuration.

Because the saved configuration is stored in non-volatile memory, it can be accessed and activated at later time.

Upon issuing the save-config command, the SBC displays a reminder on screen stating that you must use the activate-config command if you want the configurations to be updated.

```
oraclesbc1 # save-config
Save-Config received, processing.
waiting 1200 for request to finish
Request to 'SAVE-CONFIG' has Finished,
Save complete
Currently active and saved configurations do not match!
To sync & activate, run 'activate-config' or 'reboot activate'.
oraclesbc1 #
```

Activating the Configuration

On issuing the activate-config command, the saved configuration is copied from the non-volatile memory to the volatile memory. The saved configuration is activated and becomes the running configuration.

Some configuration changes are service affecting when activated. For these configurations, the SBC warns that the change could have an impact on service with the configuration elements that will potentially be service affecting. You may decide whether or not to continue with applying these changes immediately or to apply them at a later time.

```
oraclesbc1# activate-config
Activate-Config received, processing.
waiting 120000 for request to finish
Request to 'ACTIVATE-CONFIG' has Finished,
Activate Complete
oraclesbc1#
```


Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200