Oracle ECB and SBC interworking with MS Teams and Multi-vendor IP-PBX Environment

Technical Application Note

ORACLE
COMMUNICATIONS

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.

Revision History

Version	Description of Changes	Date Revision Completed
1.0	Oracle ECB and SBC interworking with MS Teams and Multi-vendor IP-PBX Environment	25th October 2021
1.1	Minor changes w.r.t formatting	12th November 2021
1.2	Removed reference to sip-all FQDN from the app note document	10th January 2022
1.3	Since sip-all FQDN is removed, add the following two sections: Enable refer call xfer on realm	22nd July 2022

Table of Contents

1. Intended Audience

This document is intended for use by Oracle Systems Engineers, third party Systems Integrators, Oracle Enterprise customers and partners and end users of the Oracle Enterprise Session Border Controller (SBC) and Oracle Enterprise Communication Broker (ECB). It is assumed that the reader is familiar with basic operations of the Oracle Enterprise Communication Broker platform along with Oracle Enterprise Session border controller platform along with on premise IP-PBX and MS Teams

2. Document Overview

This Oracle technical application note outlines how to use the Oracle ECB features to interwork between on premise IP-PBX and MS Teams. The solution contained within this document has been tested using Oracle Communication SBC with **software version 840p7** version and Oracle Communication ECB **with software version 330p1**.

This application note has been tested with **Oracle ECB and SBC with Microsoft Teams Media Bypass - Enterprise Model with Local Media Optimization (LMO) enabled**. There are other models of MS teams like Oracle ESBC with Microsoft Teams Media Bypass - Enterprise Model, Oracle SBC with Microsoft Teams Carrier Model, Oracle ESBC with Microsoft Teams Non Media Bypass - Enterprise Model, Oracle SBC with Microsoft Teams Survivable Branch Appliance (SBA) and Integration of Oracle SBC with Analog Devices and Microsoft Teams Direct Routing which customers can use based on their needs and the reference to these models can be found in the below link under “Microsoft Teams” Section.

https://www.oracle.com/technical-resources/documentation/acme-packet.html

In addition, it should be noted that the ECB configuration provided in this guide focuses strictly on the on premise IP- PBX and MS Teams. Many ECB and 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.

Please note that the IP address, FQDN and config name and its details given in this document is used as reference purpose only. The same details cannot be used in customer config and the end users can use the configuration details according to their network requirements. Also, the configuration of on premise IP-PBX and MS teams are out of scope of this document. There are some public facing IPs (externally routable IPs) that we use for our testing are masked in this document for security reasons. The customers can configure any publicly routable IPs for these sections as per their network architecture needs.

Please refer the below app notes given as an example for other configuration which is not covered on this app note for on premise PBX (CUCM/Avaya/Genesys) / MS Teams with Oracle SBC/ECB.

The actual config may differ somewhat but these docs can be only used as a reference.

https://www.oracle.com/a/otn/docs/avaya-with-ms-teams-integration-with-verizon-trunk.pdf

https://www.oracle.com/a/otn/docs/ms-teams--sbc-ecb-with-cucm-integration-v1.1.pdf

https://www.oracle.com/a/otn/docs/avaya-remote-worker-with-tls.pdf

https://www.oracle.com/a/otn/docs/oracle_sbc_with_genesys_sip_server_remote_worker.pdf

3. Introduction

3.1. Audience

This is a technical document intended for telecommunications engineers with the purpose of configuring Oracle ECB specific features with Oracle SBC interworking with on premise IP-PBX and MS Teams. There will be steps that require navigating the ECB GUI interface and Oracle SBC GUI interface. Understanding the basic concepts of TCP/UDP, IP/Routing, DNS server and SIP/RTP are also necessary to complete the configuration and for troubleshooting, if necessary.

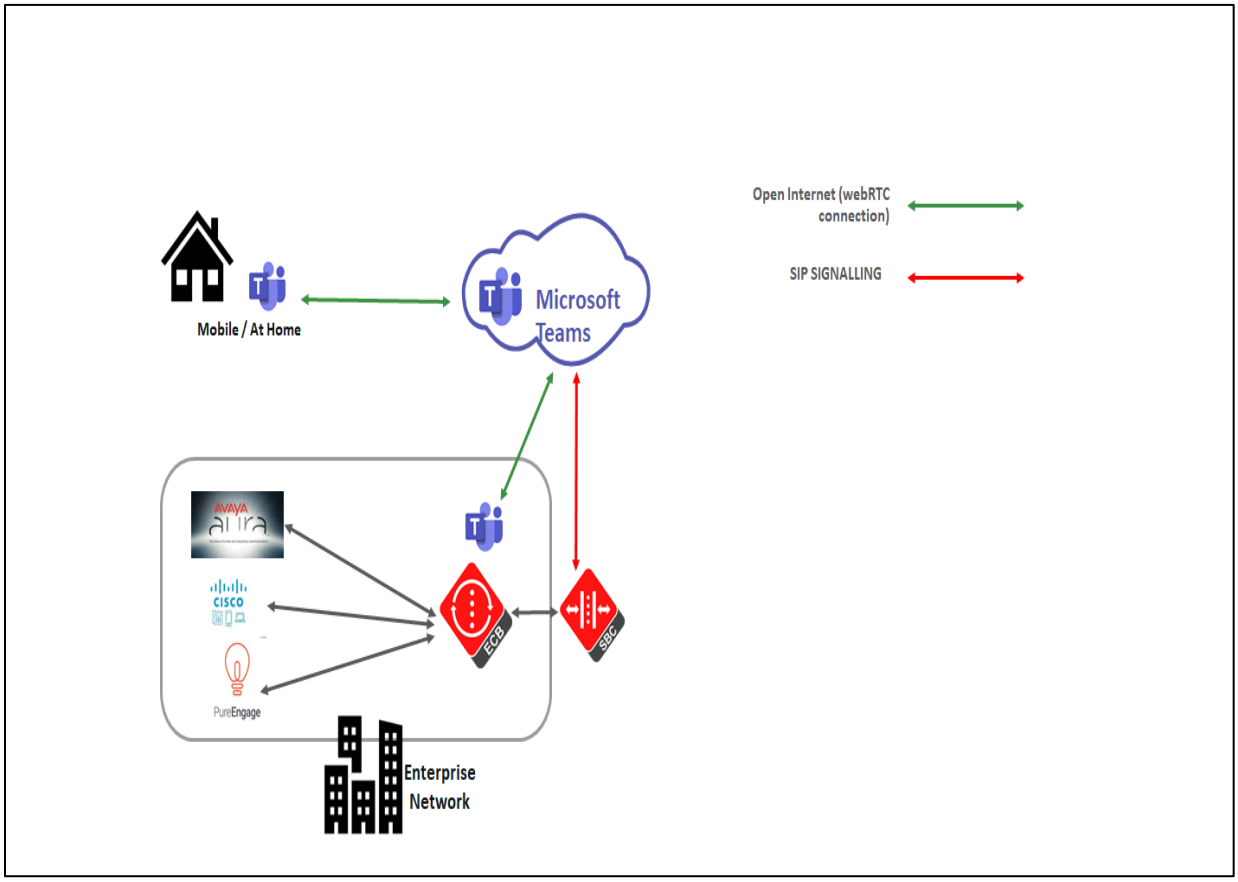
3.2. Requirements

- Oracle Enterprise Session Border Controller (hereafter Oracle SBC) running 8.4.0 version
- Oracle Enterprise Communication Broker (hereafter Oracle ECB) running 3.3.0 version

The below revision table explains the versions of the software used for each component:
This table is Revision 1 as of now:

Software Used	SBC Version	ECB Version
Revision 1	8.4.0	3.3.0

3.3. Architecture



The configuration, validation and troubleshooting is the focus of this document and will be described in two phases:

- Phase 1 – Configuring the Oracle ECB.
- Phase 2 – Configuring the Oracle SBC.

4. New ECB Configuration

The Oracle ECB is available either as an appliance or as an application for operation on virtual machines. When running as an appliance, the Oracle ECB software is packaged with the Netra Server X3-2 and delivered to the end customers. When running as a virtual application, the Oracle ECB software can be deployed on any third-party COTS hardware that meets the specified guidelines.

Once the ECB is deployed (in the appliance mode or the application mode) and connected, you can power on the ECB. Software installation of the ECB is required upon first startup. Although the Oracle ECB is primarily configured through the GUI, you need to perform the software installation and certain steps via the CLI. For our testing, we will first configure certain common configuration and we will do some feature specific configuration after that specific to our testing.

4.1. ECB CLI initial config

Power on the ECB and confirm that you see the following output from the boot-up sequence.

The default username for the User level is “user” and the default password is “acme”.
The default username for an Administrator level is “admin”, and the default password is "packet".
Both passwords have to be changed according to the rules shown below.

```
Password:
%
% Only alphabetic (upper or lower case), numeric and punctuation
% characters are allowed in the password.
% Password must be 8 - 64 characters,
% and have 3 of the 4 following character classes :
%      - lower case alpha
%      - upper case alpha
%      - numerals
%      - punctuation
%
Enter New Password:
Confirm New Password:

Password is acceptable.
```

Now set the management IP of the ECB by setting the IP address in bootparam

To access bootparam. Go to Configure terminal->bootparam.

Note: There is no management IP configured by default.

```
LabECB#
LabECB# conf t
LabECB(configure)# bootparam

'.' = clear field;  '-' = go to previous field;  q = quit

Boot File               : /boot/nnPCZ330p1.bz
IP Address              : 10.138.194.175
VLAN                    :
Netmask                 : 255.255.255.192
Gateway                 : 10.138.194.129
IPv6 Address            :
IPv6 Gateway            :
Host IP                 :
FTP username            :
FTP password            :
Flags                   :
Target Name             : LabECB
Console Device          : VGA
Console Baudrate        : 115200
Other                   :

NOTE: These changed parameters will not go into effect until reboot.
Also, be aware that some boot parameters may also be changed through
PHY and Network Interface Configurations.



LabECB(configure)#
LabECB(configure)#
```

Setup product type to Enterprise Communication broker as shown below.

To configure product type, type in setup product in the terminal

```
LabECB#
LabECB#
LabECB# setup product

------------------------------------------------------------
WARNING:
Alteration of product alone or in conjunction with entitlement
changes will not be complete until system reboot

Last Modified 2017-02-03 09:44:20
------------------------------------------------------------
 1 : Product       : Enterprise Communication Broker

Enter 1 to modify, d' to display, 's' to save, 'q' to exit. [s]: 1
```

Enable the features for the ECB using the setup entitlements command as shown

Save the changes and reboot the ECB.

```
LabECB#
LabECB# show entile
LabECB# show entil
LabECB# show entitlements
Provisioned Entitlements:
-------------------------
Enterprise Communication Broker Base    : enabled
Session Capacity                        : 10000

Keyed (Licensed) Entitlements
-----------------------------

LabECB#
```

Go to configure terminal->system->http-server-config.

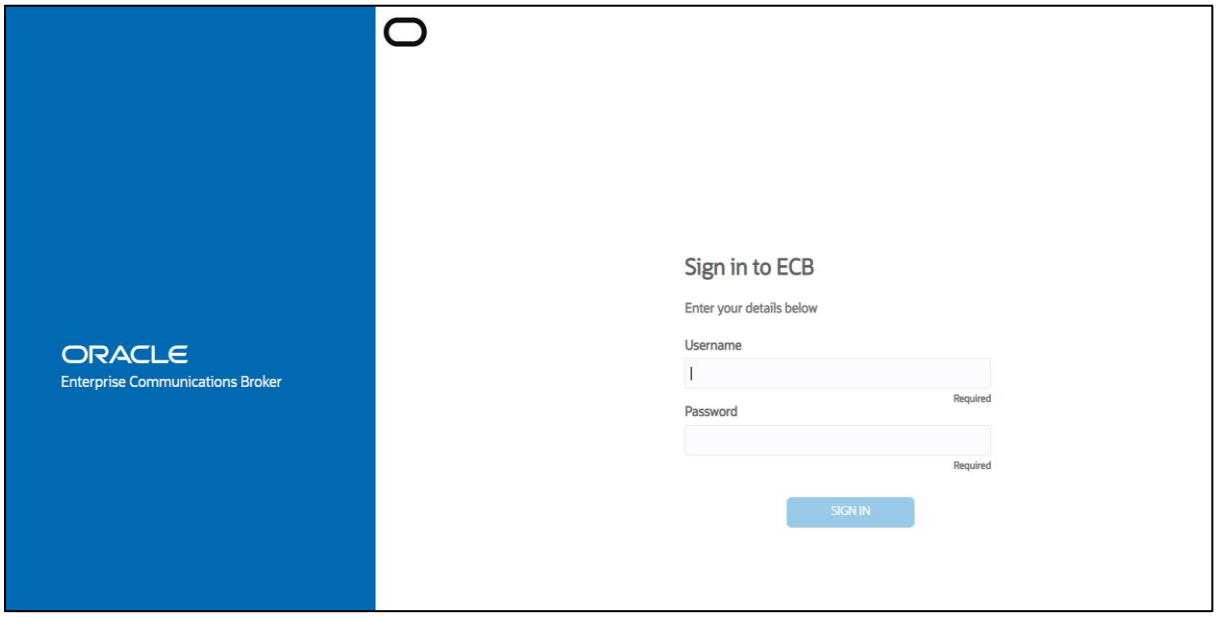
Enable the http-server-config to access the ECB using Web GUI. Save and activate the config.

```
LabECB(http-server)# show
http-server
        name                                    webServerInstance
        state                                   enabled
        realm
        ip-address
        http-state                              enabled
        http-port                               80
        https-state                             disabled
        https-port                              443
        http-interface-list                     GUI
        http-file-upload-size                   0
        tls-profile
        last-modified-by                        @
        last-modified-date                      2021-09-16 21:07:32

LabECB(http-server)#
LabECB(http-server)#
LabECB(http-server)#
```

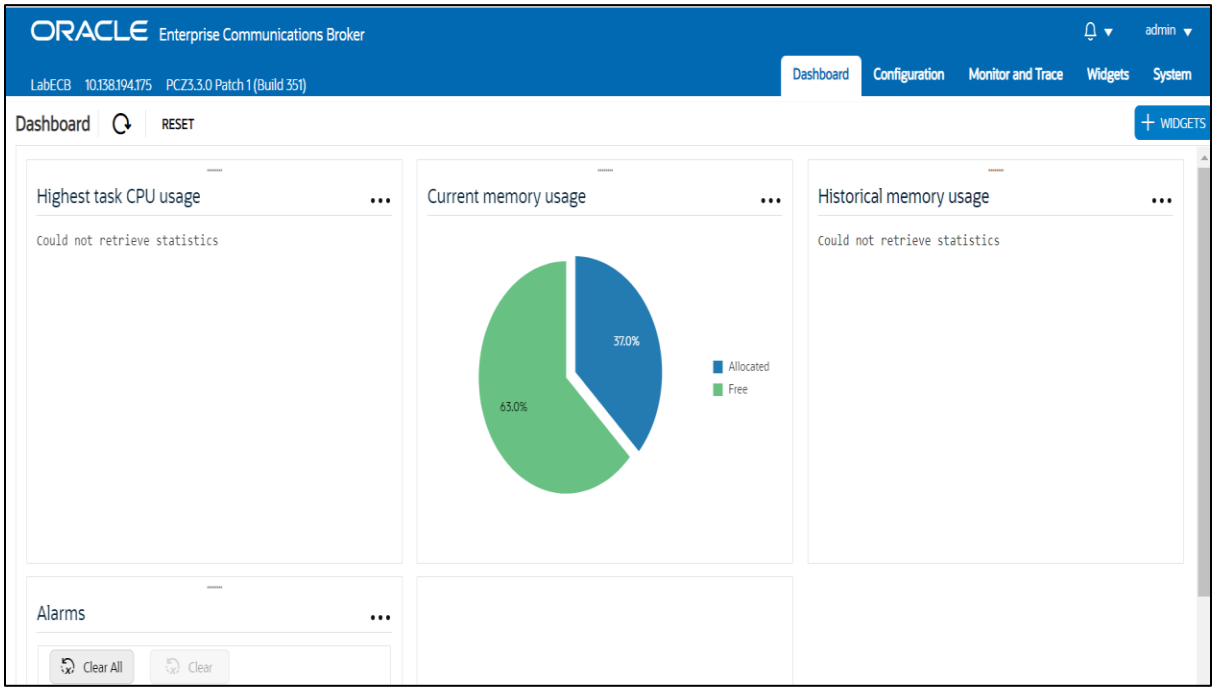
4.2 Logging into the ECB

You can now access the ECB through the Web GUI.
Start an Internet browser and start the GUI using the URL: http://server IP address/.
The login screen will appear.

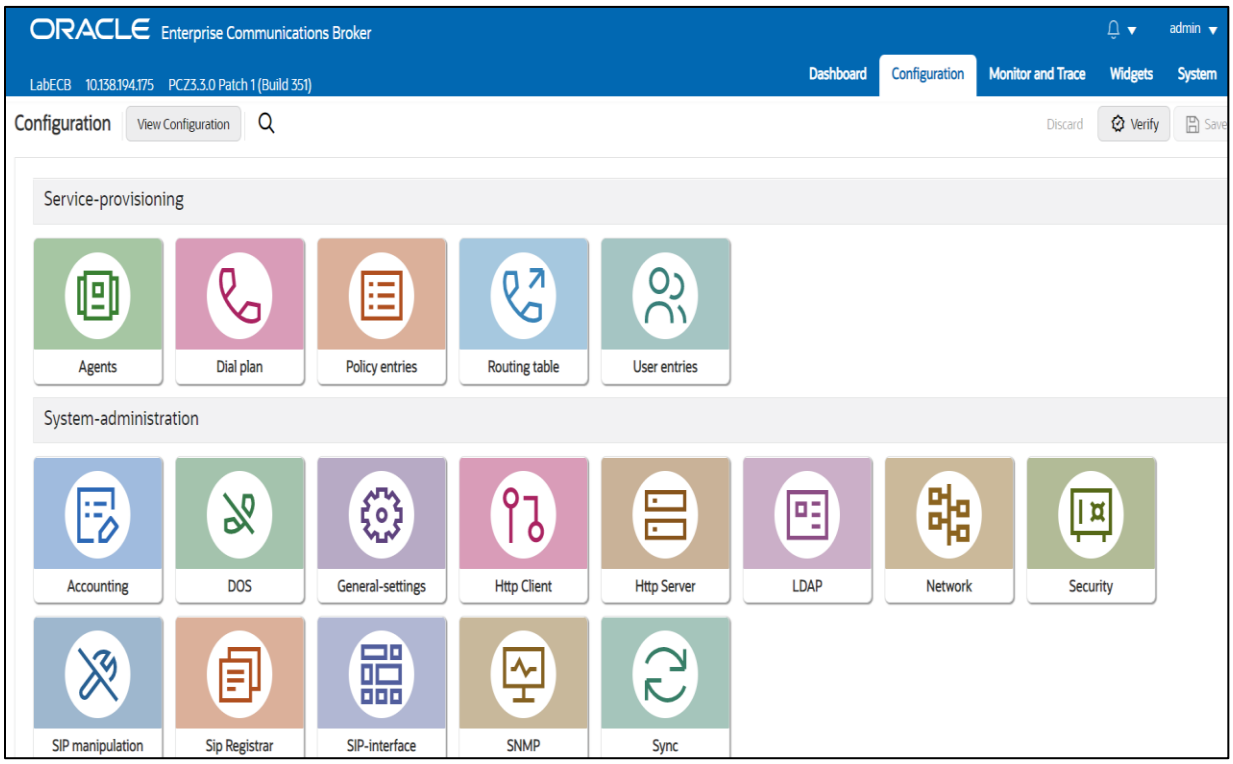


Enter the username and password and this is same as CLI username & password.

After logging into the ECB, the Home screen will be displayed.
The Oracle ECB GUI has the screen shown below:

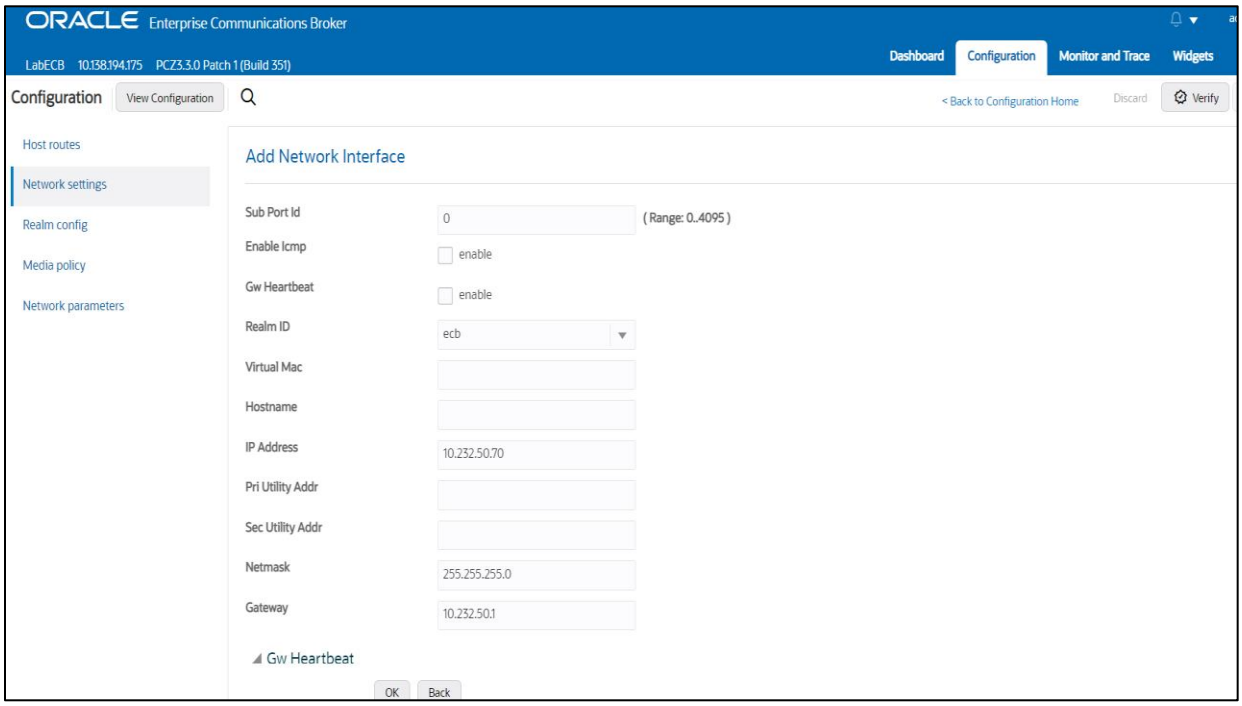


Please click Configuration tab and the tab appears as shown below:



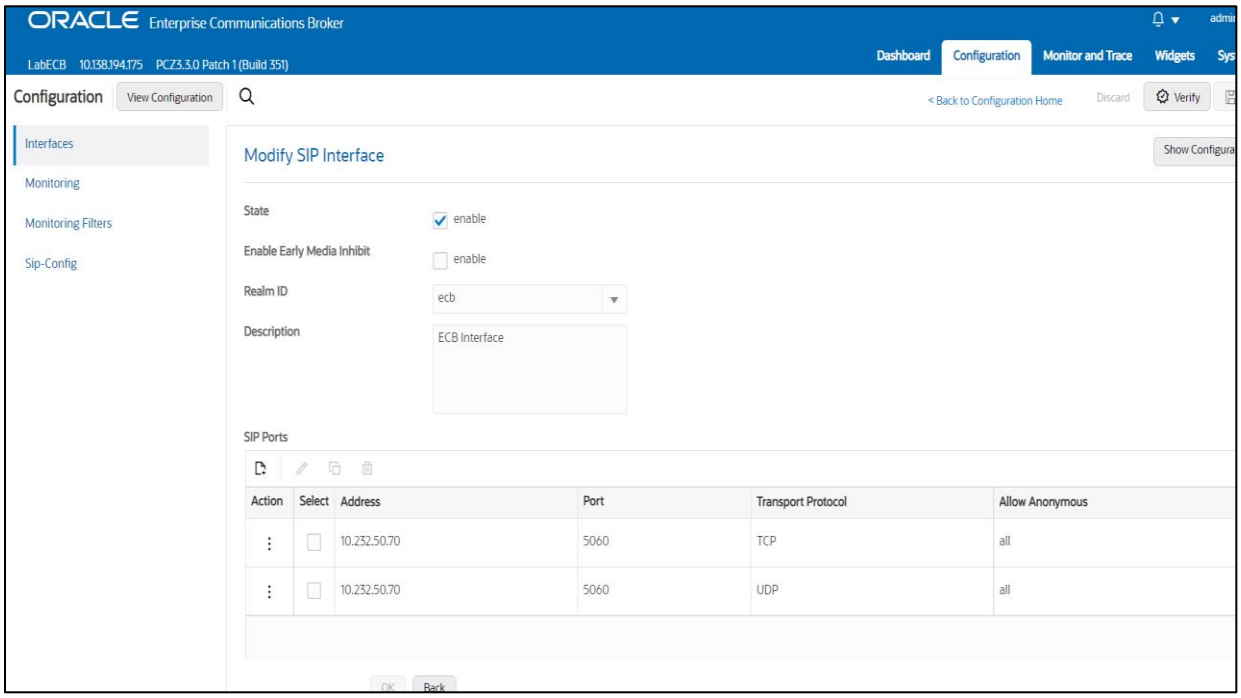
4.3. Add Network Settings

Go to System Administration ---- Network --- Network Settings and Click Add



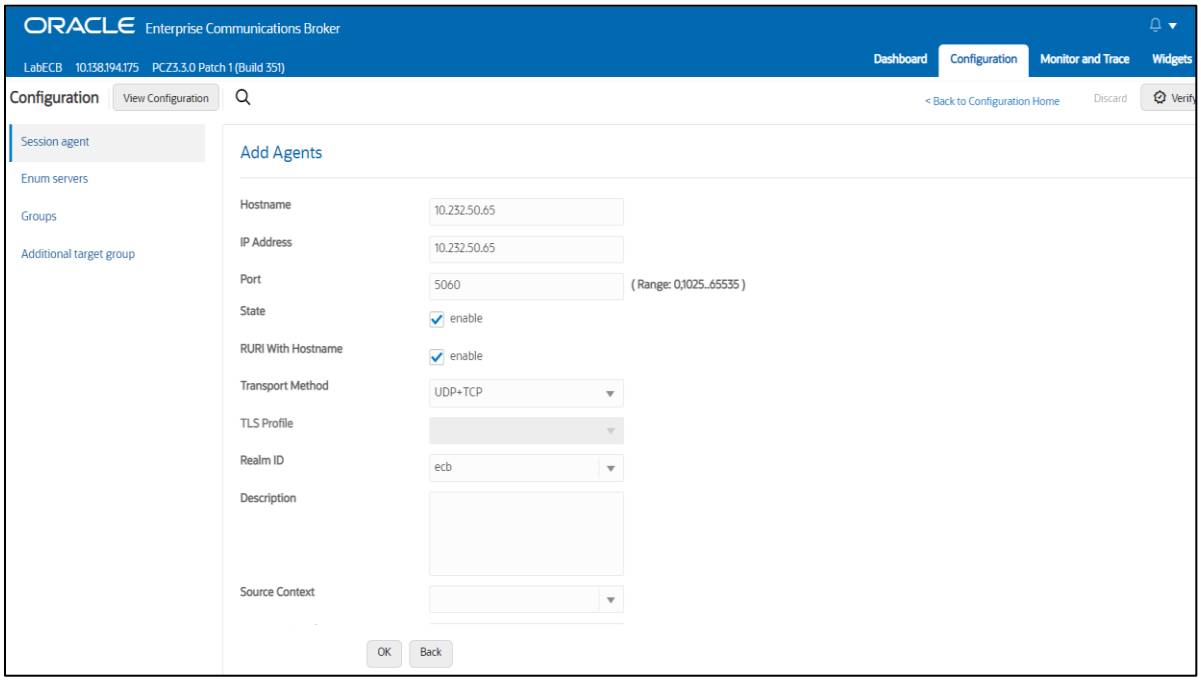
4.4. Configure SIP Interface

Go to System Administration ---- SIP Interfaces --- Interfaces and Click Add

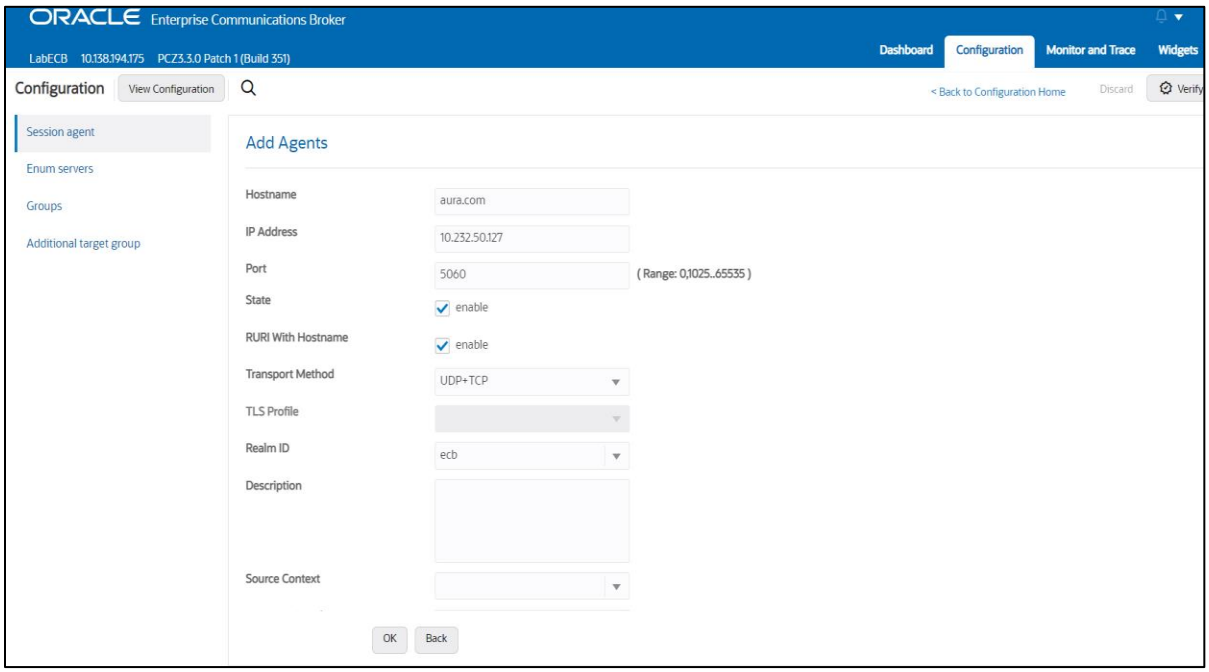


4.5. Configuring the Agents

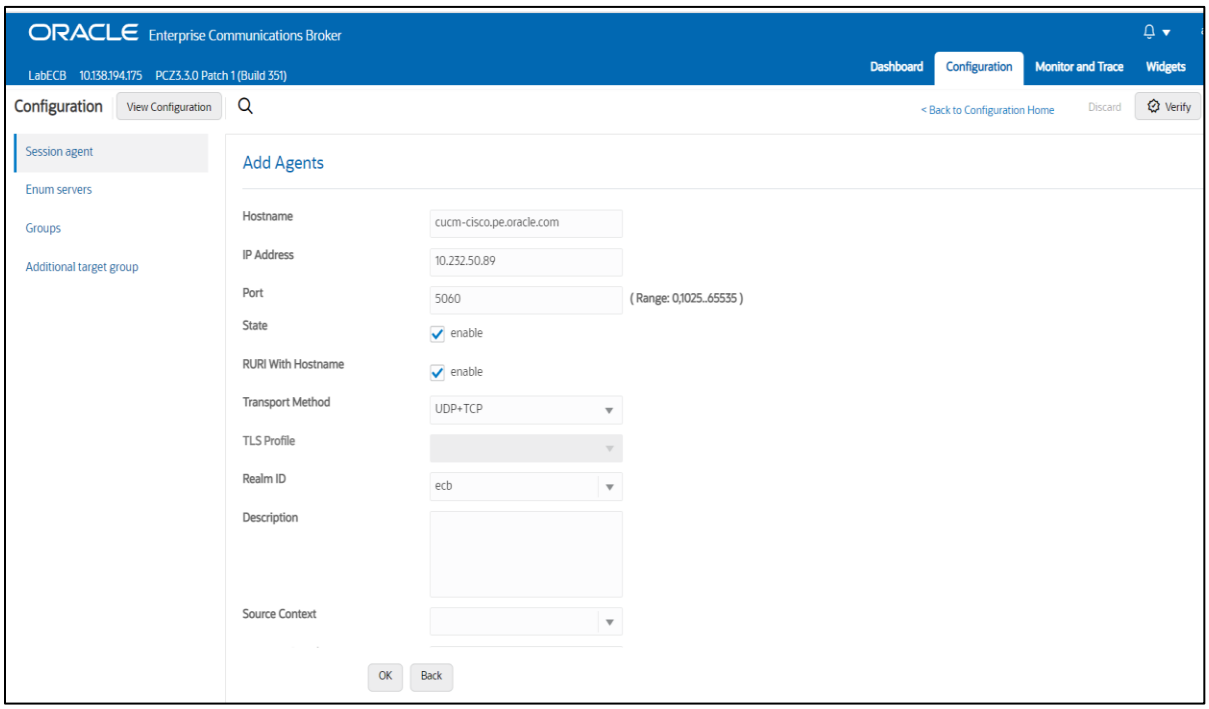
Click Configuration --- Service Provisioning ----- Agents --- Session Agents and Click Add.
We will now add 10.232.50.65 as Agent to ECB (SBC SIP interface).
This will be used to connect to PBX like Teams through SBC from the ECB.



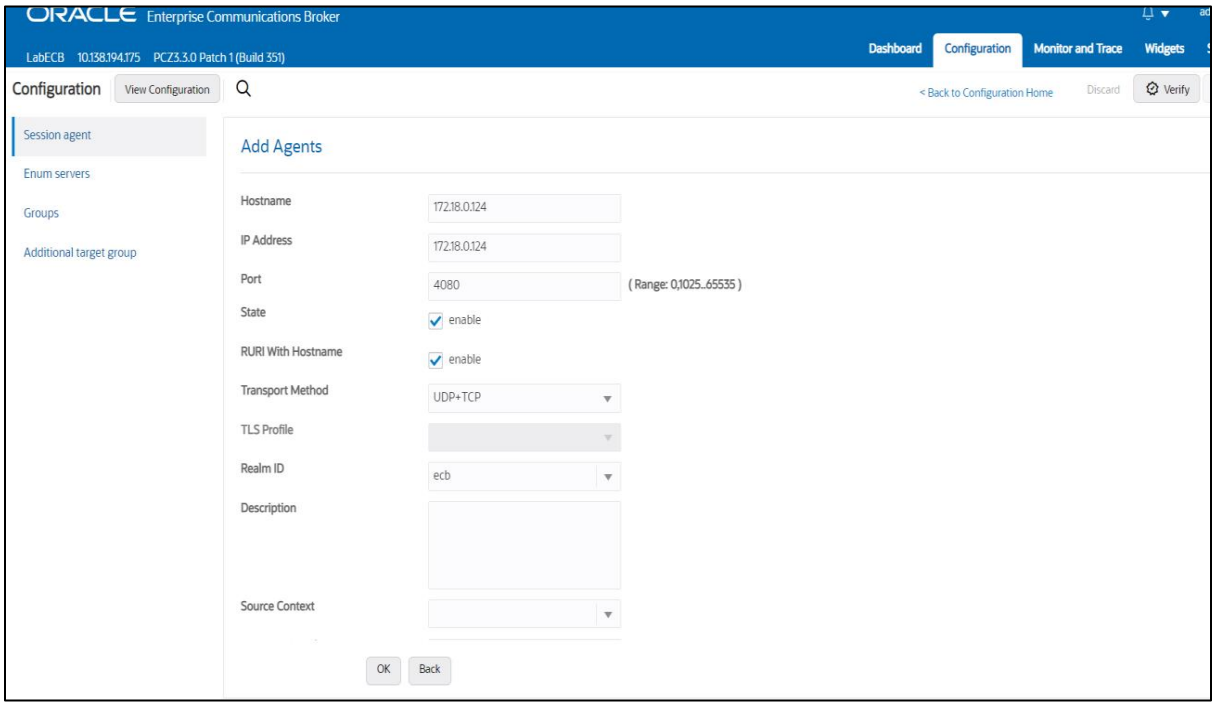
Similarly, add other IP-PBX (Core Side) as agents to ECB like given below.
First add 10.232.50.127 (Avaya server) to the ECB.



We can next add 10.232.50.89 or cucm-cisco.pe.oracle.com (Cisco CUCM) to the ECB.



Finally, we can add 172.18.0.124 (Genesys Pure Engage) to the ECB.



Please keep the parameter **"Egress Number Translation Mode = E164-no-plus**" for all Session Agent configured above in the ECB. With this, ECB common configurations are complete.
We will proceed to feature specific configuration from the next section.

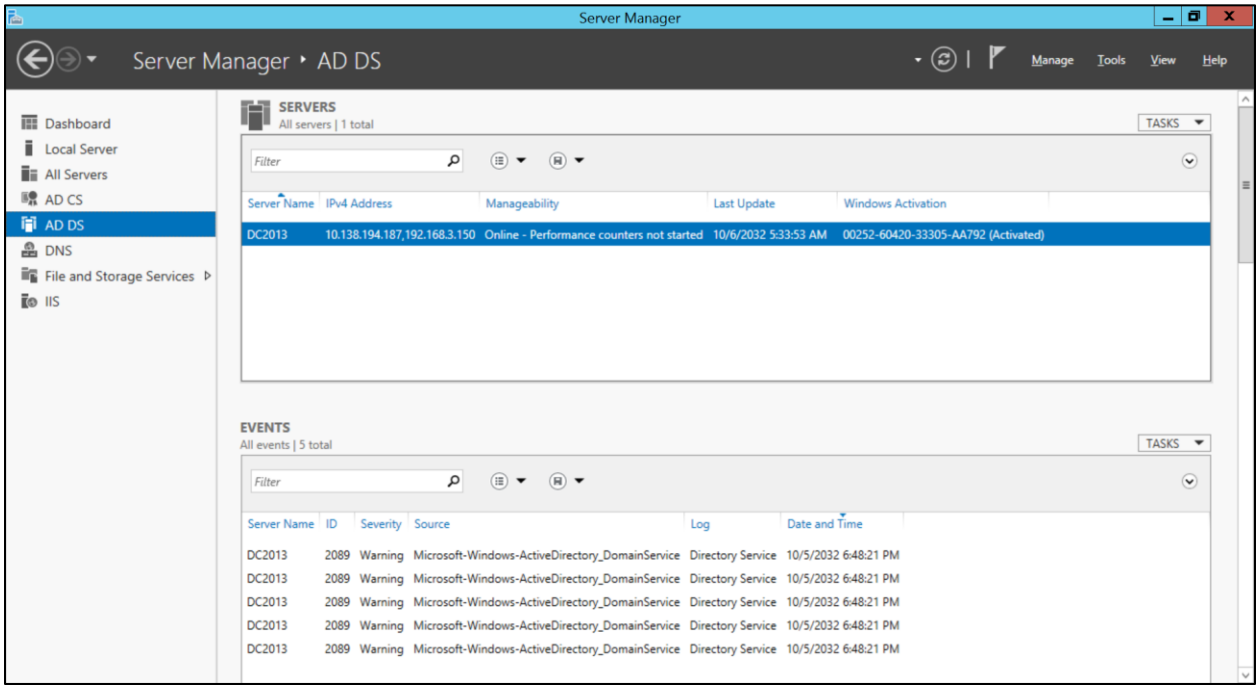
4.6. ECB with LDAP configuration

To test the Oracle ECB with LDAP authentication, we have used the Windows AD configuration with ECB. Regarding AD server, we need AD with Domain controller Service (AD DS) installed with which we can add Users to the AD which can be queried using LDAP protocol. If we configure windows AD with only LDS service (AD LDS), we will not be able to add Users in Active Directory and hence we cannot use AD server for LDAP configuration.

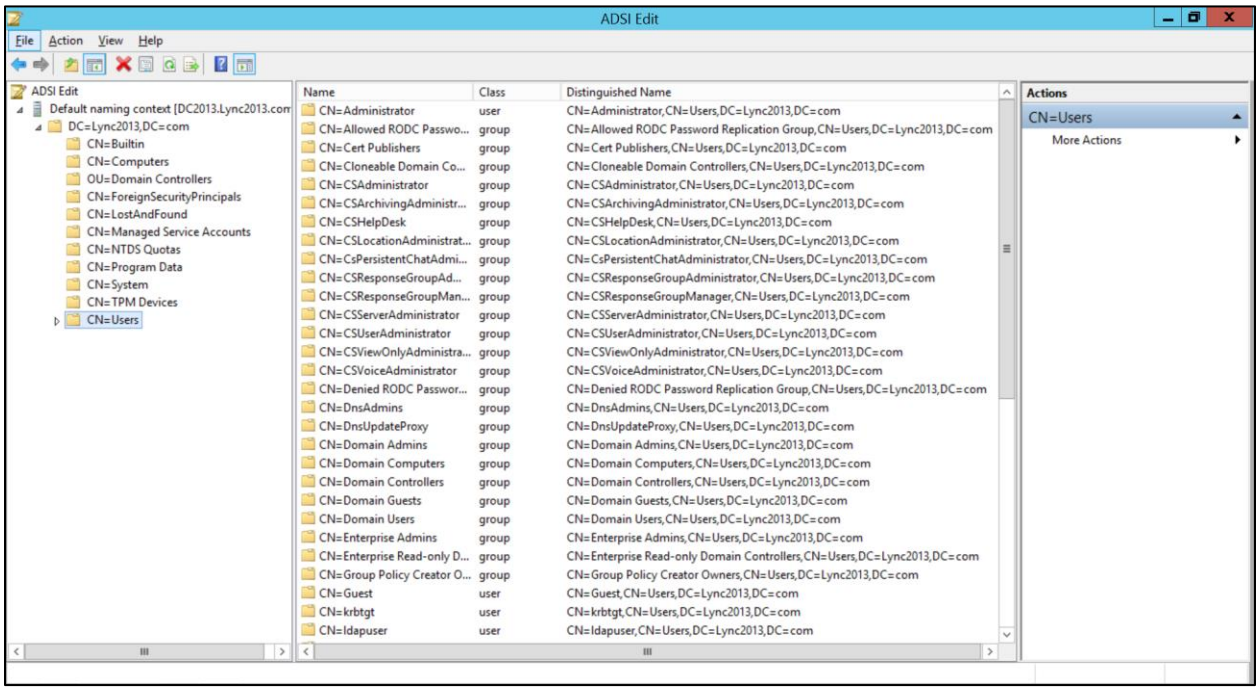
The Windows AD DS server is configured in the lab with 10.138.194.187 IP and Administrator/Abcd1234 credentials.
For more information on how to configure Windows Server with AD server and using ADSI and other option, please refer to below link

https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/deploy/install-active-directory-domain-services--level-100-

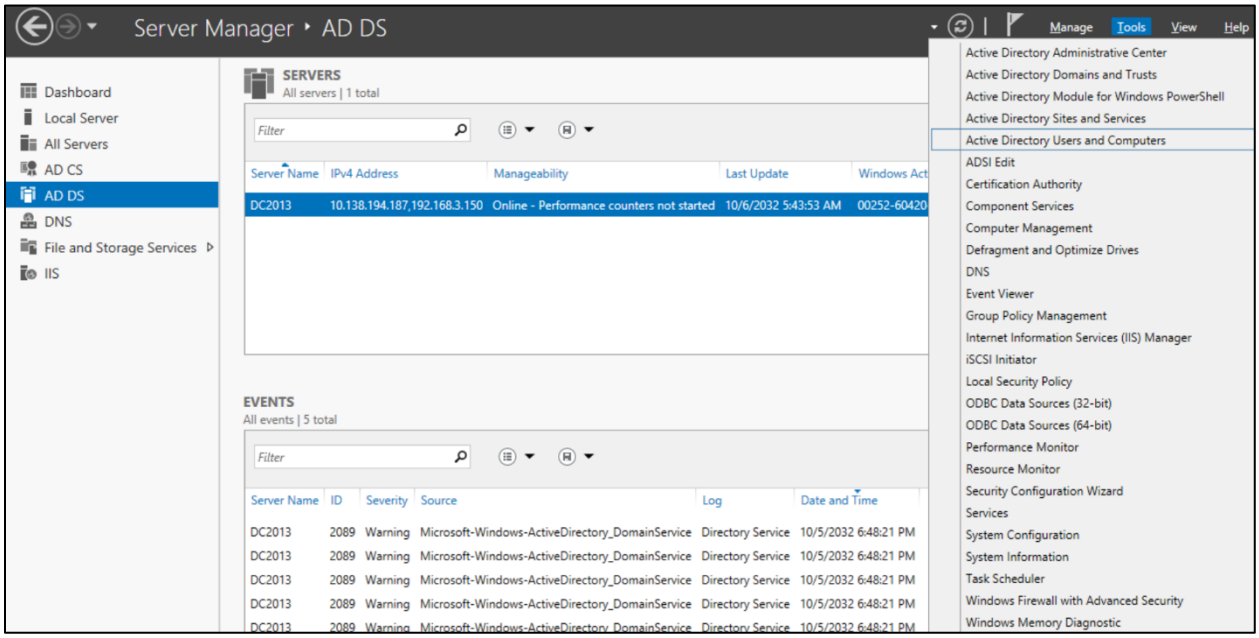
The Screenshot from Windows AD side is given below: (High Level)

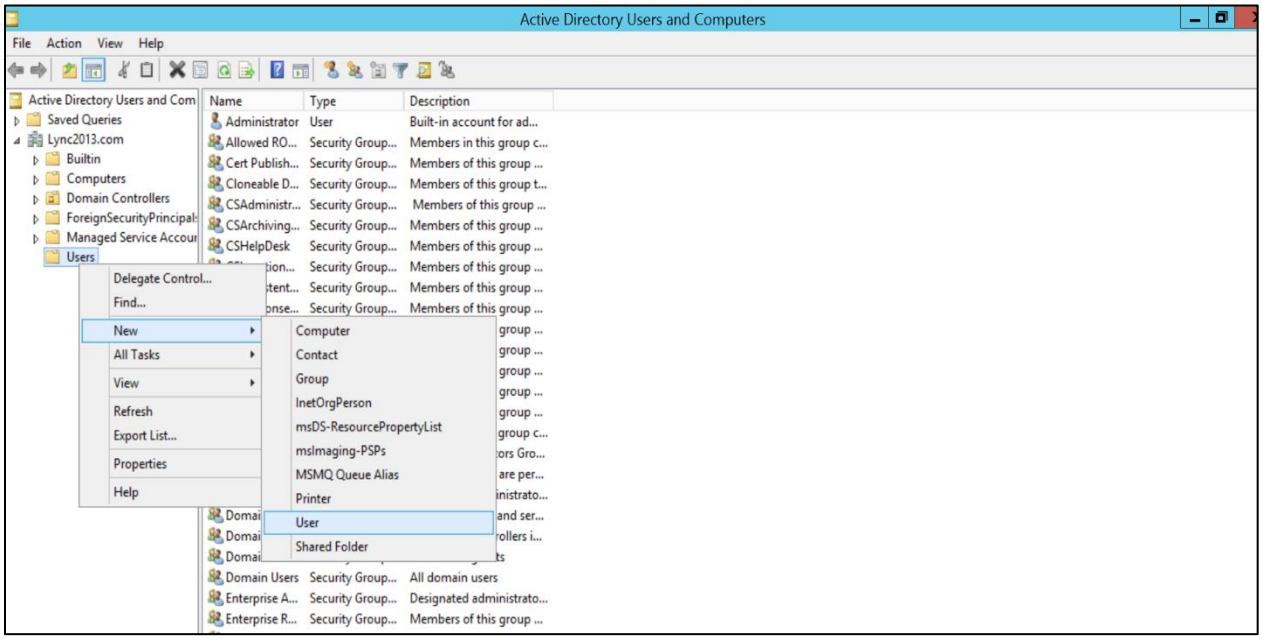


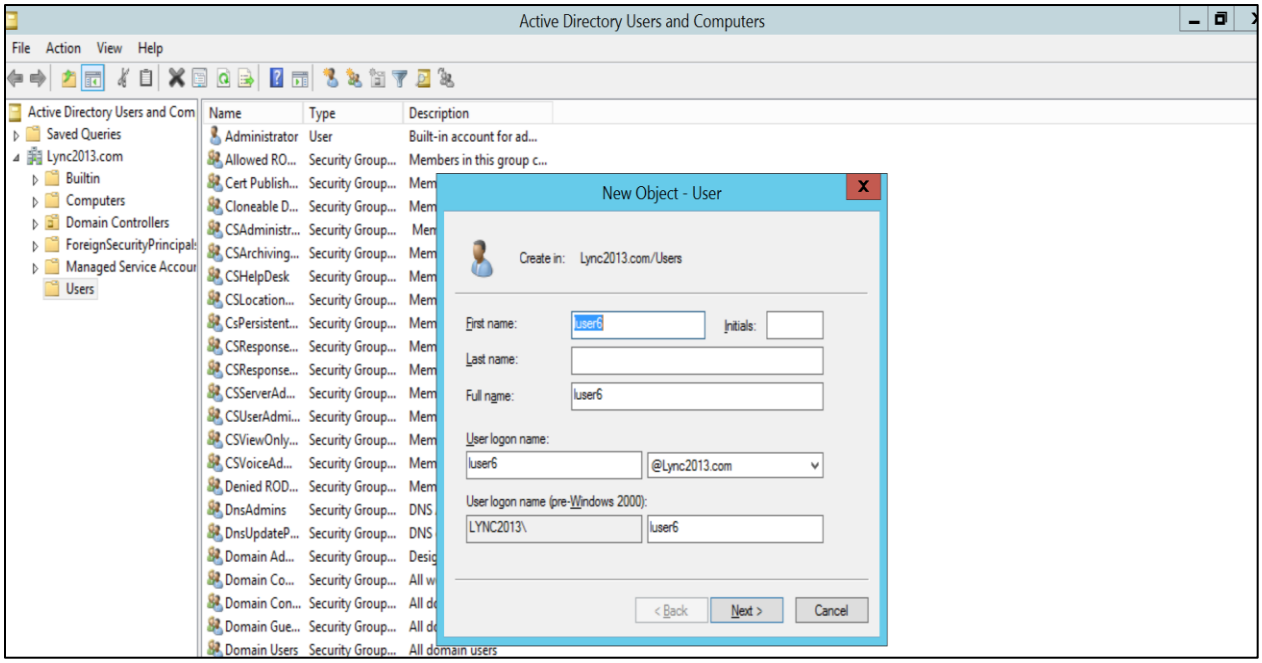
Right Click the AD server and you will get various options and please select ADSIEdit option to open it and you will get the below tree structure if AD DS is configured properly.



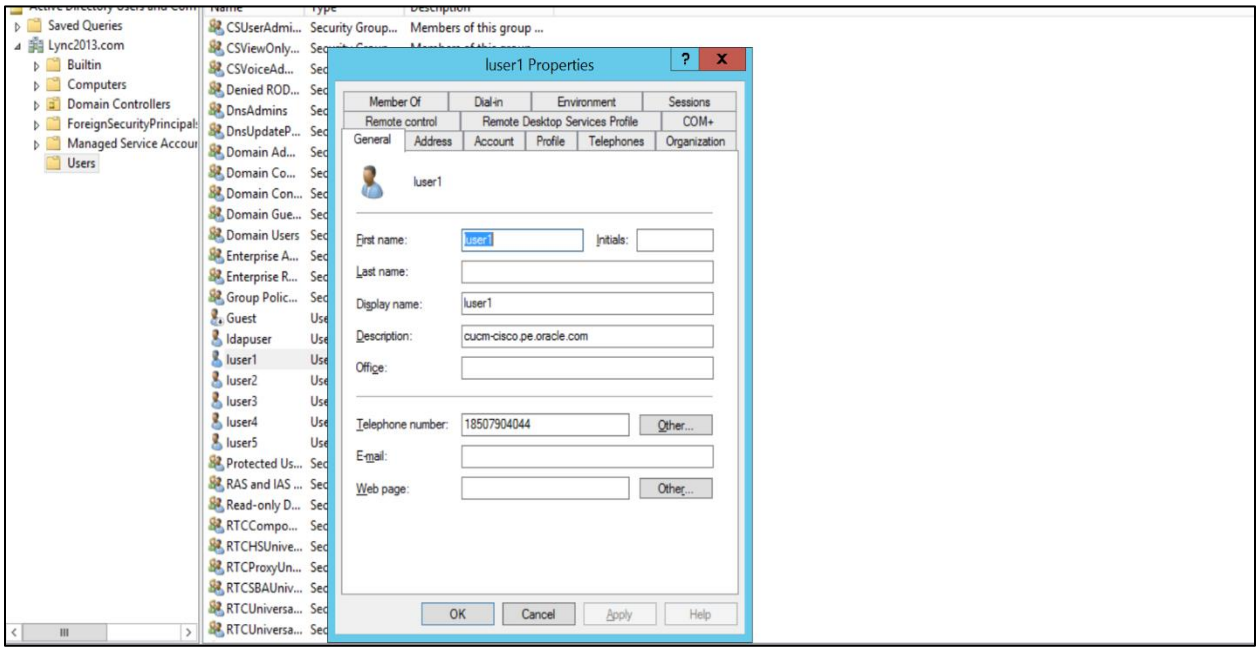
Please select AD server manager ---- tools and select users and computers tab and you can create the users that you will be querying using ECB LDAP configuration.




Active Directory Users and Computers
File Action View Help
Active Directory Users and Com
Saved Queries
Lync2013.com
Builtin
Computers
Domain Controllers
ForeignSecurityPrincipals
Managed Service Accour
Users
Name Type Description
Administrator User Built-in account for ad...
Allowed RO... Security Group... Members in this group c...
Cert Publish... Security Group... Members of this group ...
Cloneable D... Security Group... Members of this group t...
CSAdministr... Security Group... Members of this group ...
CSArchiving... Security Group... Members of this group ...
CSHelpDesk Security Group... Members of this group ...
Delegate Control...
Find...
New
All Tasks
View
Refresh
Export List...
Properties
Help
Computer
Contact
Group
InetOrgPerson
msDS-ResourcePropertyList
msImaging-PSPs
MSMQ Queue Alias
Printer
User
Shared Folder
Domain Users Security Group... All domain users
Enterprise A... Security Group... Designated administrato...
Enterprise R... Security Group... Members of this group ...


Active Directory Users and Computers
File Action View Help
Active Directory Users and Com
Saved Queries
Lync2013.com
Builtin
Computers
Domain Controllers
ForeignSecurityPrincipals
Managed Service Accour
Users
Name Type Description
Administrator User Built-in account for ad...
Allowed RO... Security Group... Members in this group c...
New Object - User
Create in: Lync2013.com/Users
First name: user6
Initials:
Last name:
Full name: luser6
User logon name:
luser6
@Lync2013.com
User logon name (pre-Windows 2000):
LYNC2013\
luser6
< Back
Next >
Cancel

We can also add the **attributes "description" and "Telephone number**" to the users which will be used as search option in ECB LDAP config under routing/lookup query.

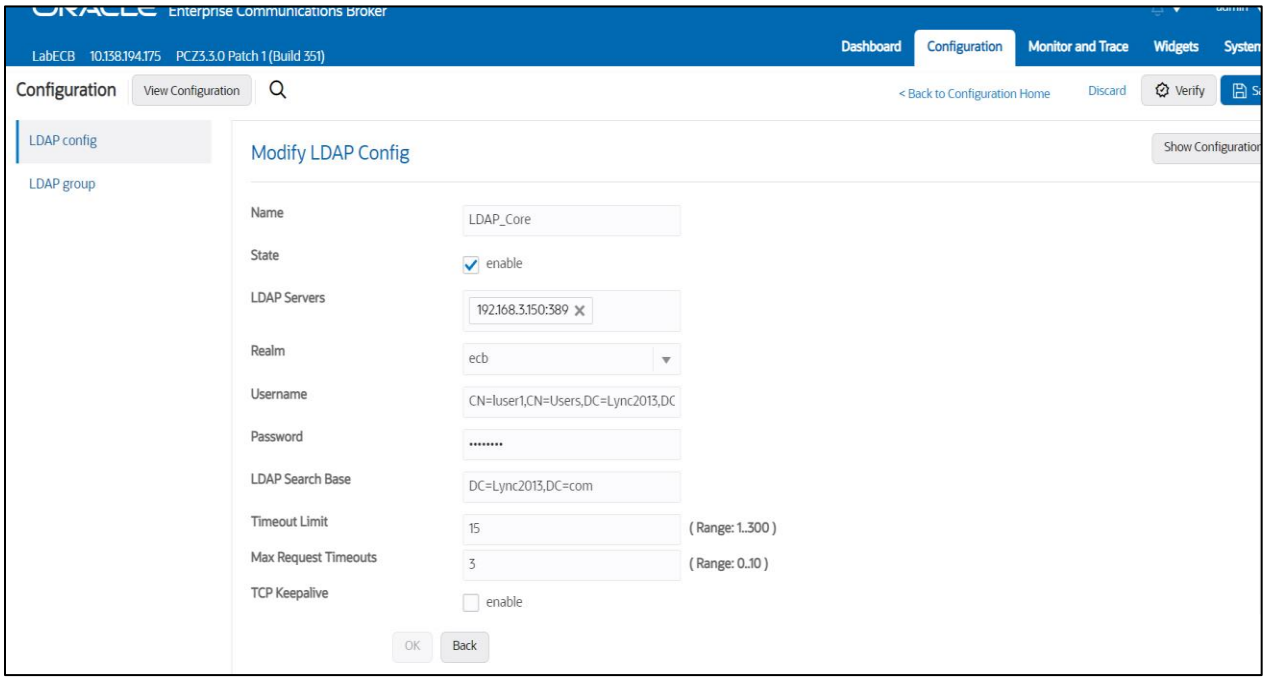


With the above steps, the user is created in the Windows AD DS server.
We can add more users using the above steps.
For our testing, we have created luser1 for Core side and luser2 for SBC side.

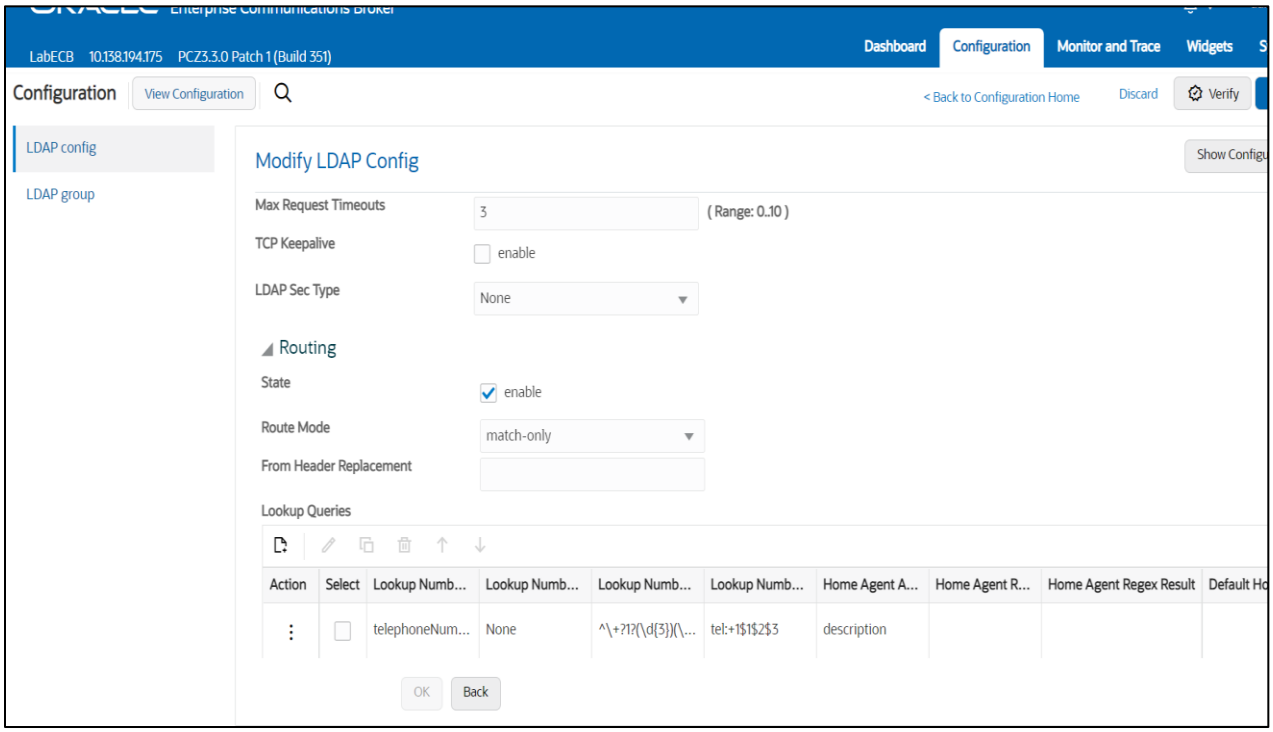
Once we have created the Users in AD DS server, we can proceed to ECB for LDAP configuration. The pre-requisite for this config to work is that we need to have Session Agents configured for both sides in the ECB.

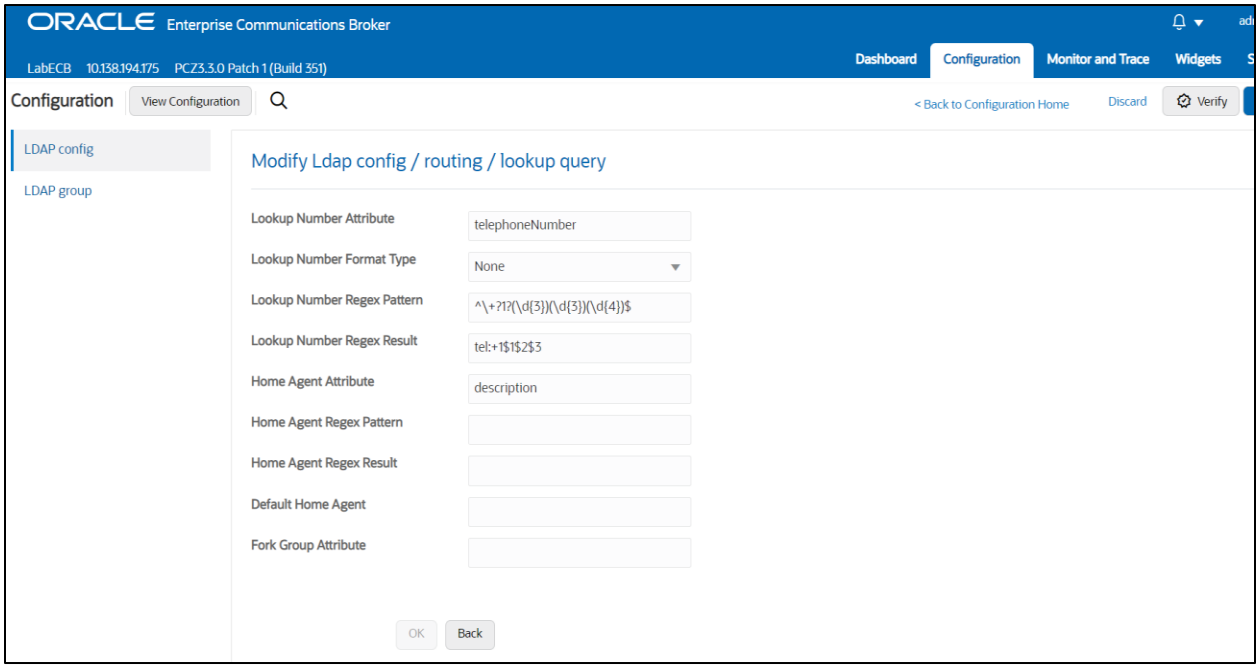
Please go to ECB and Navigate to System Administration – LDAP --- LDAP Config and do the configuration as per the testing needs. In the below example, we have created configuration for LDAP to search for core side user and SBC side user. As the Call from ECB is routed to SBC side and vice versa, SBC interface IP configured for ECB side as Session Agent in ECB.

The config given below is to search core side user. (Avaya/CUCM/Genesys).
Here in the below example, we used LDAP search for CUCM user.

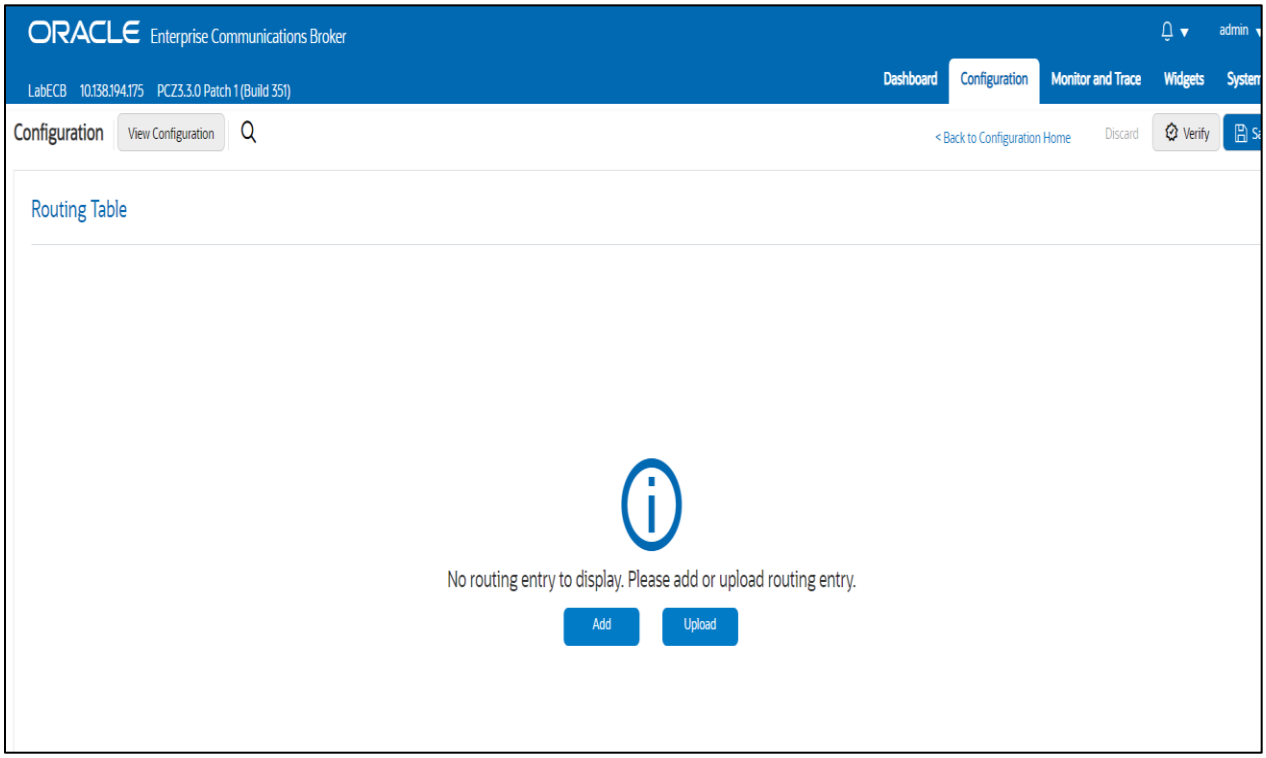


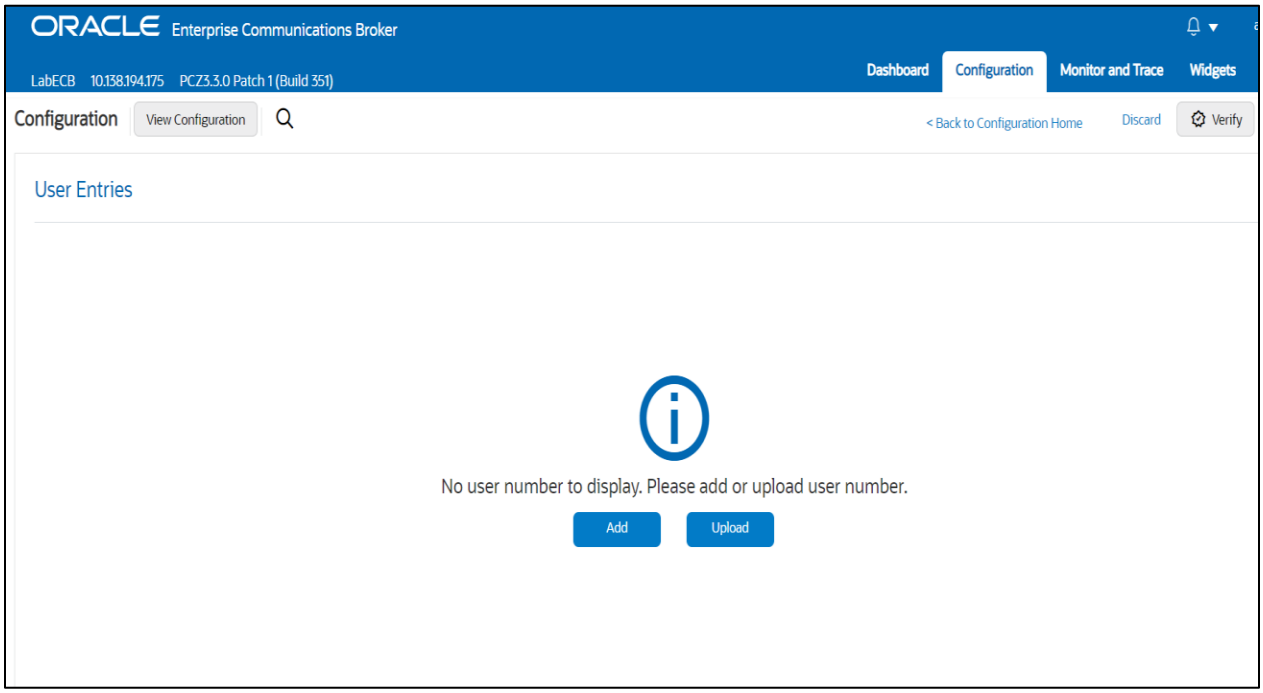
The routing looks for attribute description and telephoneNumber added in the ldap user already.





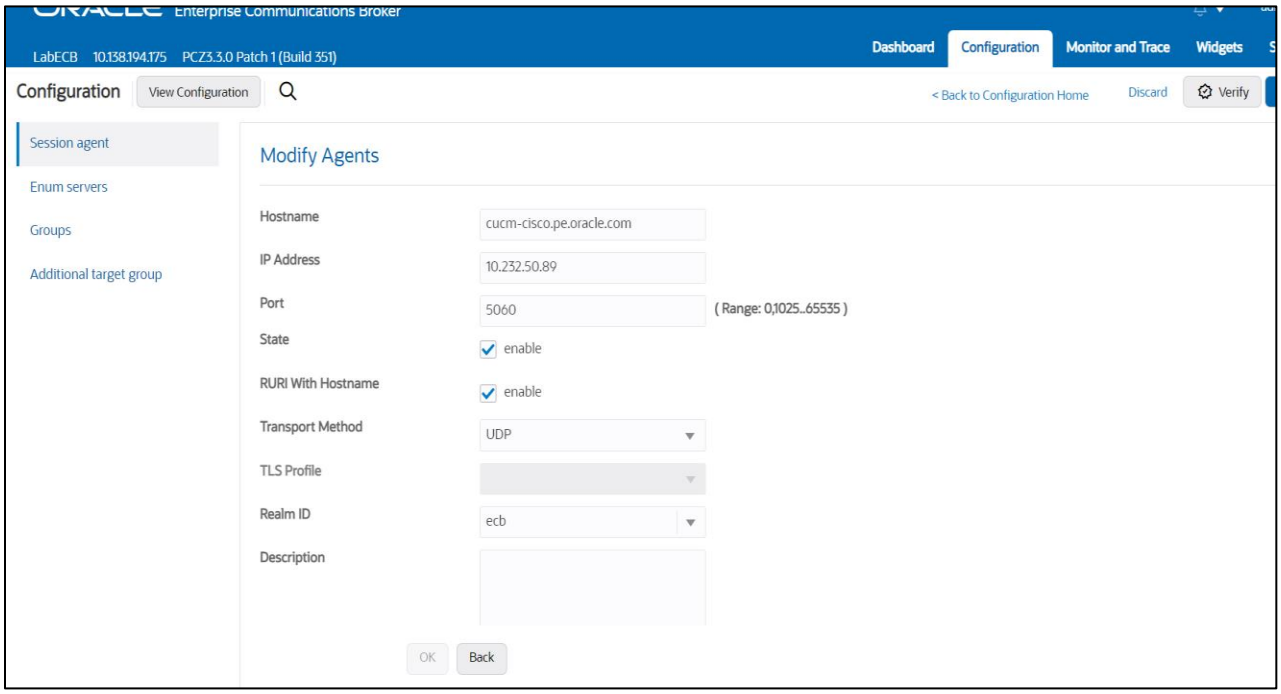
Similarly, create an LDAP config similar to above config for SBC side.
Please check that there should not be any routing config or User Entry config in ECB when we are using LDAP config as the ECB need to search only LDAP to get the next step.

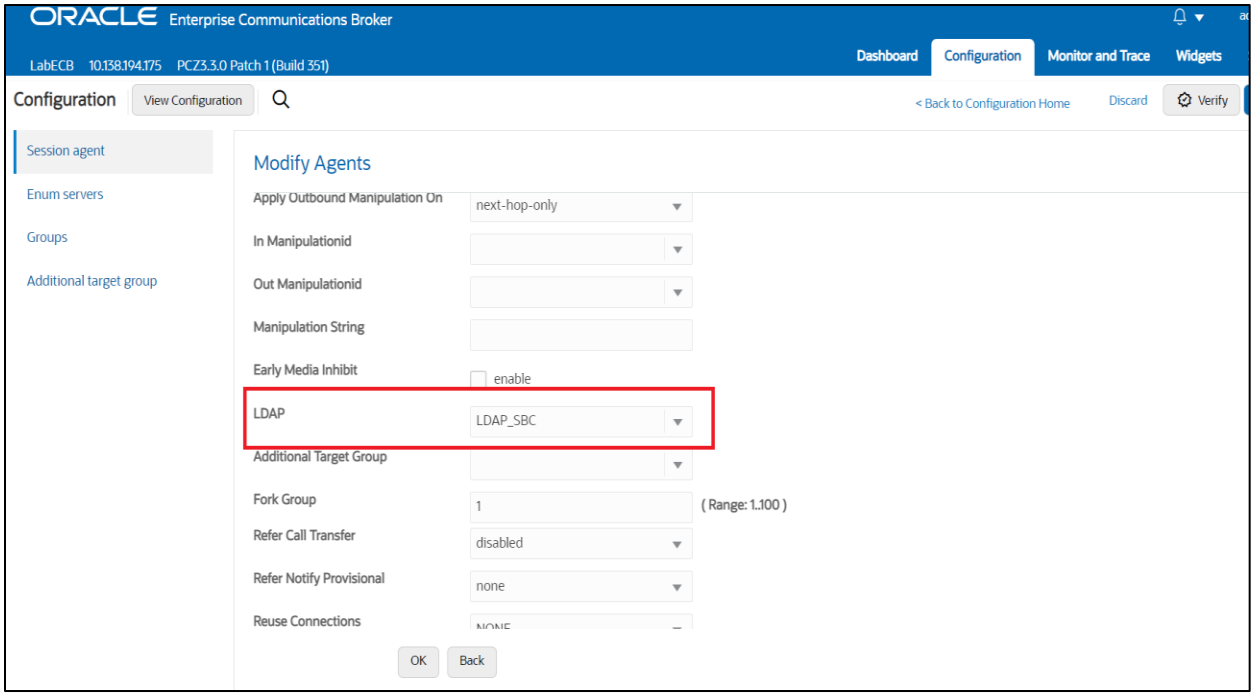




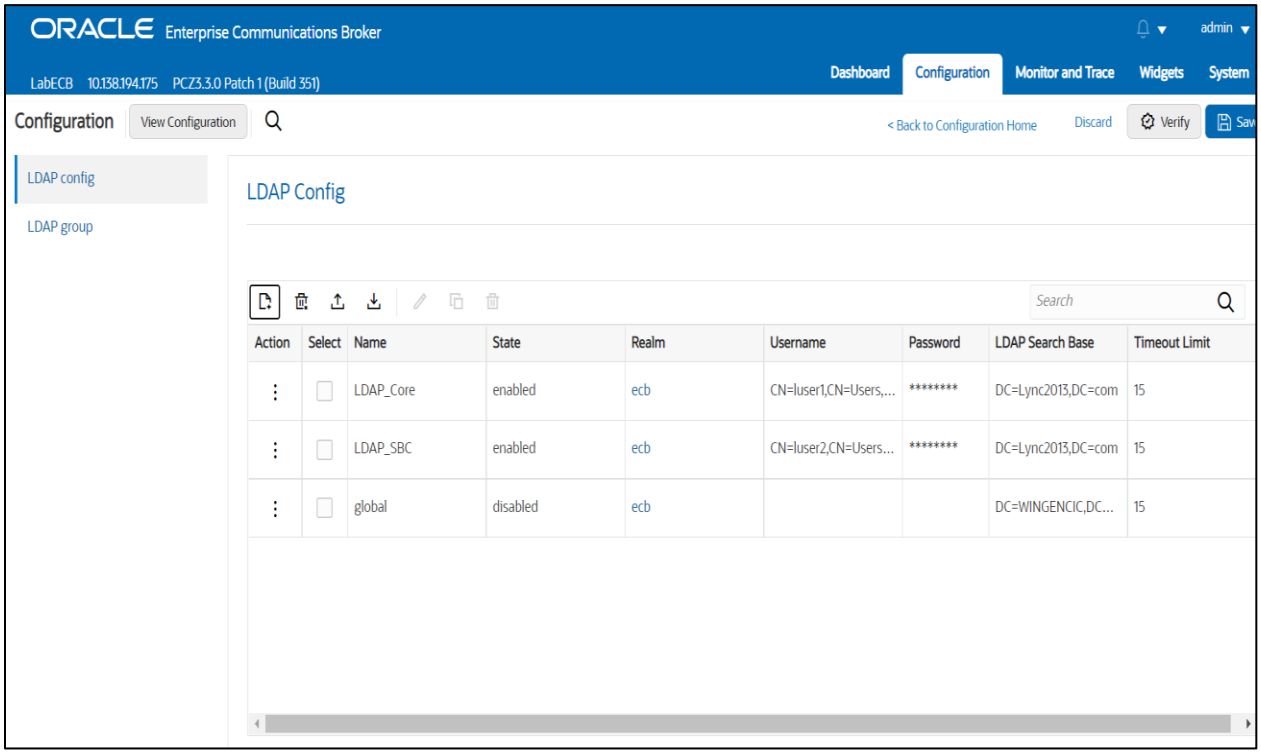
Once the config is done, please navigate to Session Agent, select the Source agent and add the LDAP config created under LDAP option as given below.

When Core side agent (Cisco CUCM) is the source side, add SBC side LDAP config under LDAP option. Save the config after making all the necessary steps.





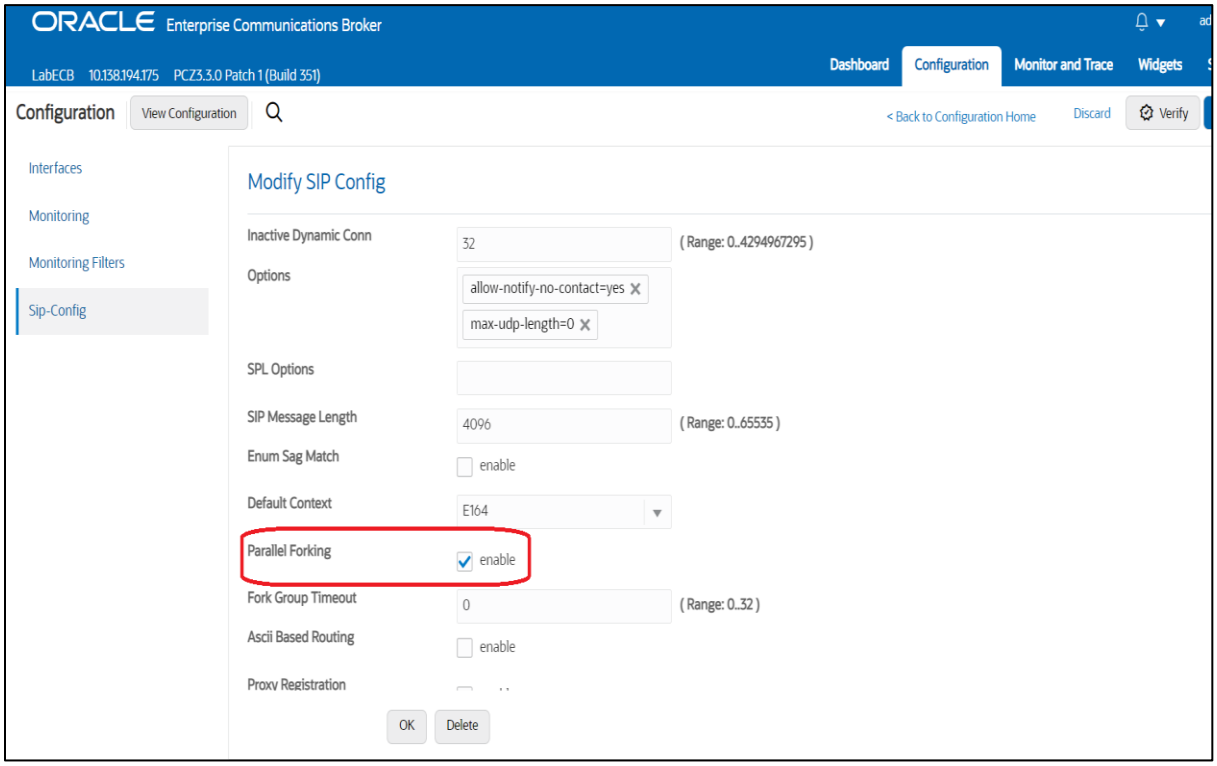
Similarly, When SBC agent is source side, add Core side LDAP config under LDAP option. The LDAP config is ECB looks like below:



4.7. ECB Parallel Forking with LDAP configuration

Oracle ECB can perform parallel forking, which directs the INVITE to all targets for an Address of Record (AOR) simultaneously. In other words, a single DID or directory number is configured in multple Agents and an inbound call needs to be simultaneously alerted to those Agents. When any target responds, the Oracle ECB issues a CANCEL to the other targets and ignores any responses from them

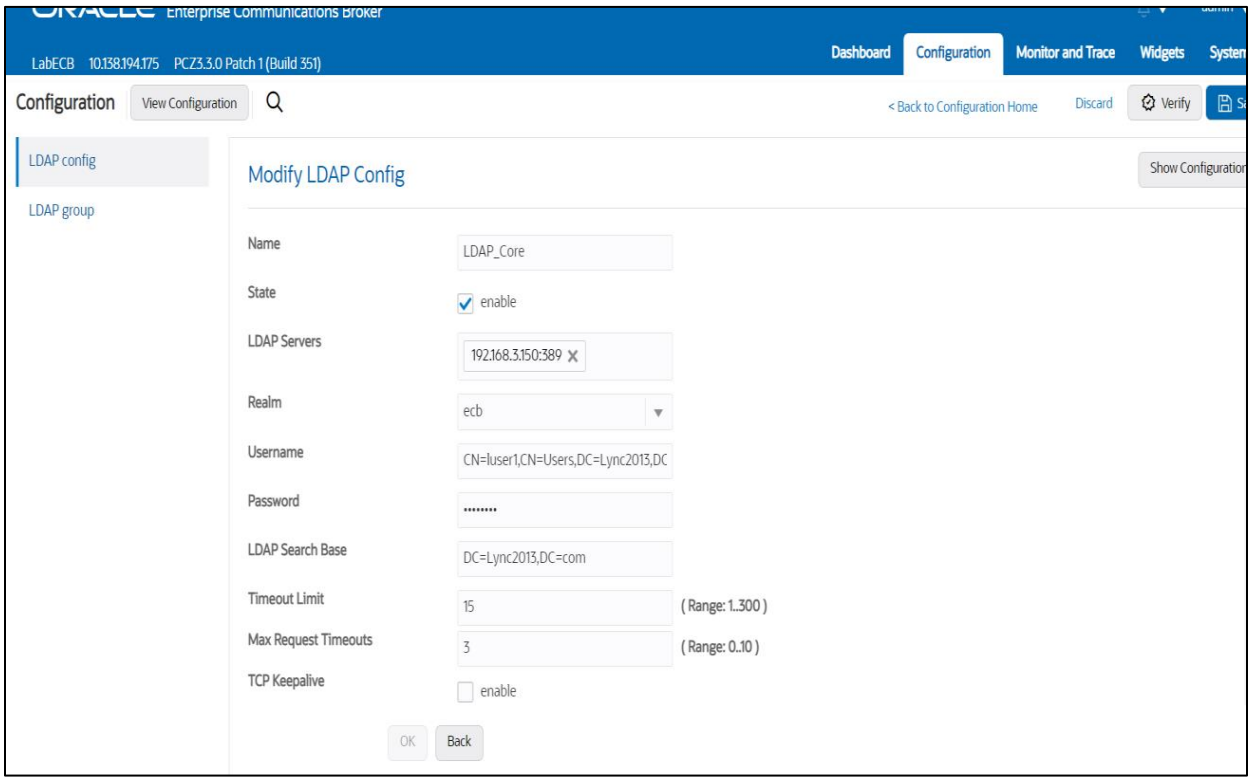
We can enable parallel forking from SIP Interface ---- SIP Config configuration of the ECB as given below.

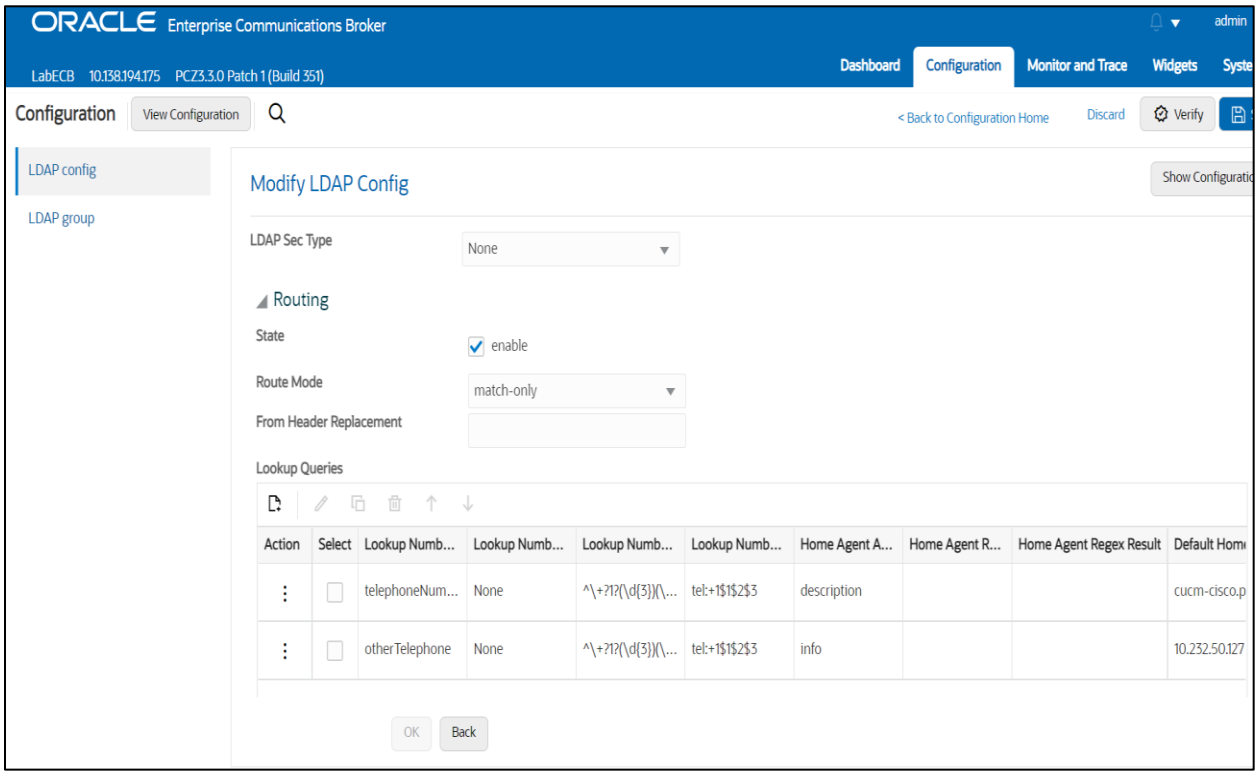


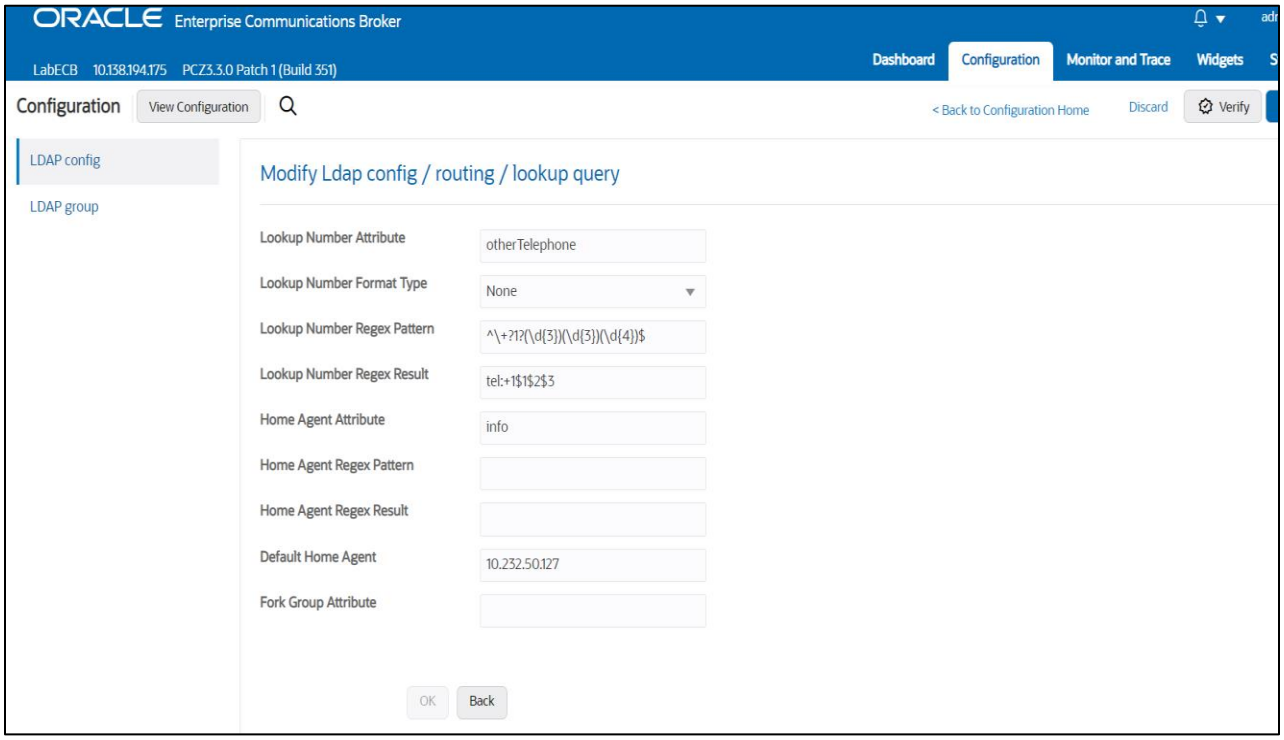
To test ECB parallel forking with LDAP config, we need to add additional attributes along with already added attributes in routing tab which is under LDAP config. The same attributes needs to be added under the LDAP user luser1 in the AD server too.

For our testing, we have added the different attribute **“otherTelephone” and “info”** which corresponds to Avaya User so that ECB sends INVITE to both PBX at same time. Save the config after that.

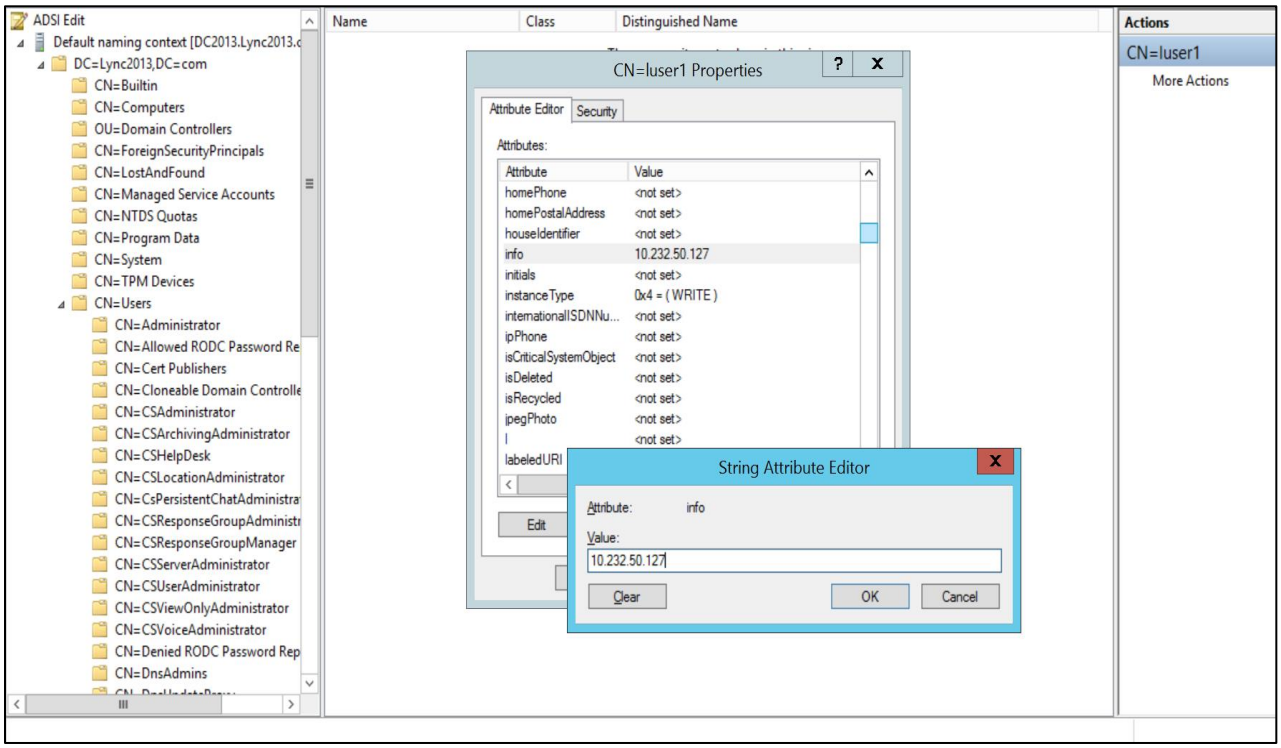
The config is shown below for ECB parallel forking with LDAP.



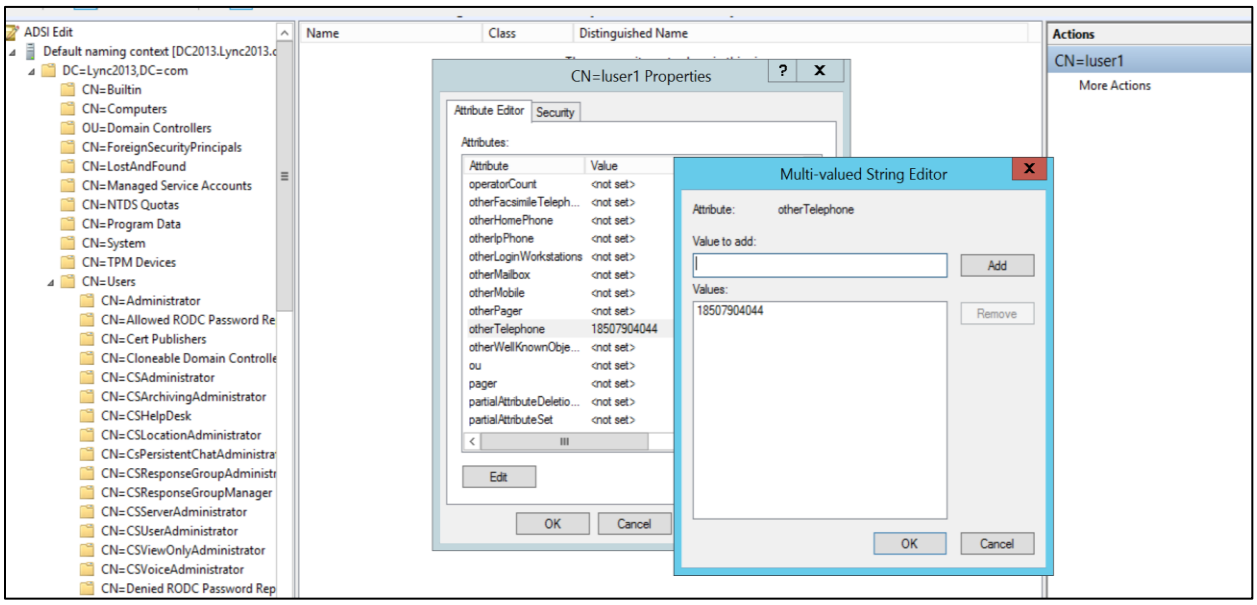




The AD user luser1 additional attribute config is given below.



The same directory number 18507904044 is given to Avaya User as well for ECB parallel forking.

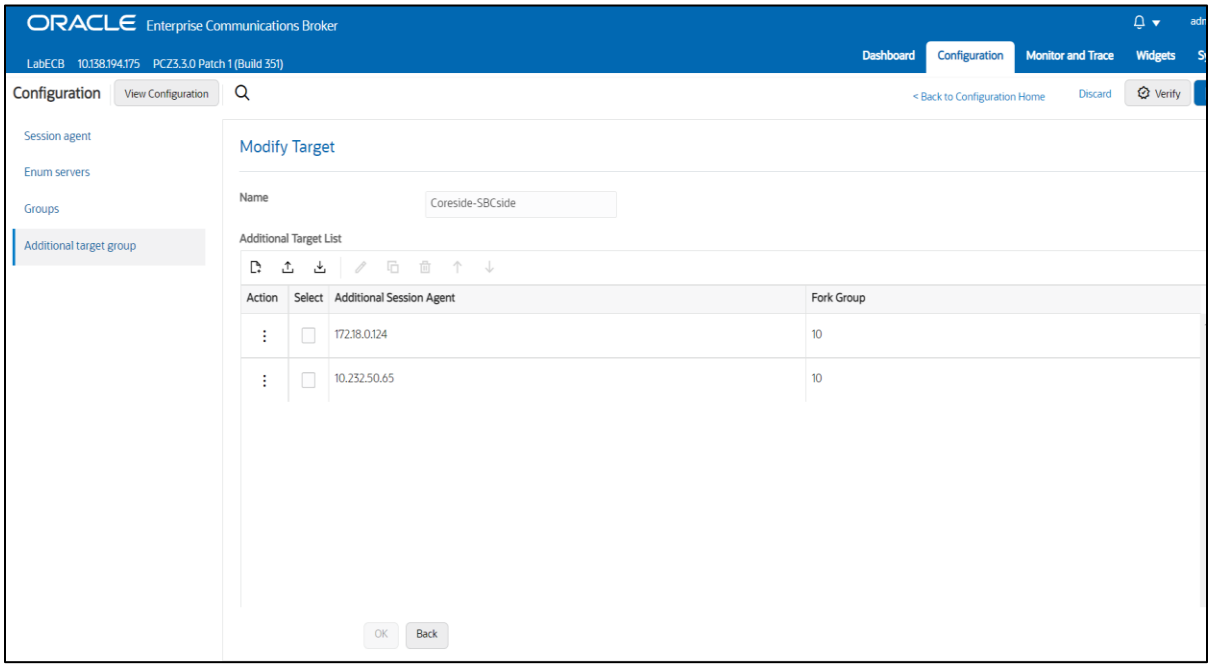


4.8. ECB Parallel Forking with Additional Target Group (ATG) configuration

Similar to LDAP parallel forking, ECB can also perform parallel forking using Additional Target Group (ATG) configuration. An Additional Target Group (ATG) is a list of agents or end stations that ECB uses as candidates for either parallel or serial forking. You can configure these ATGs with fork group numbers, which the system then uses to do forking (For our testing, we do parallel forking). The configuration for ECB ATG parallel forking is performed in different way and is given below.

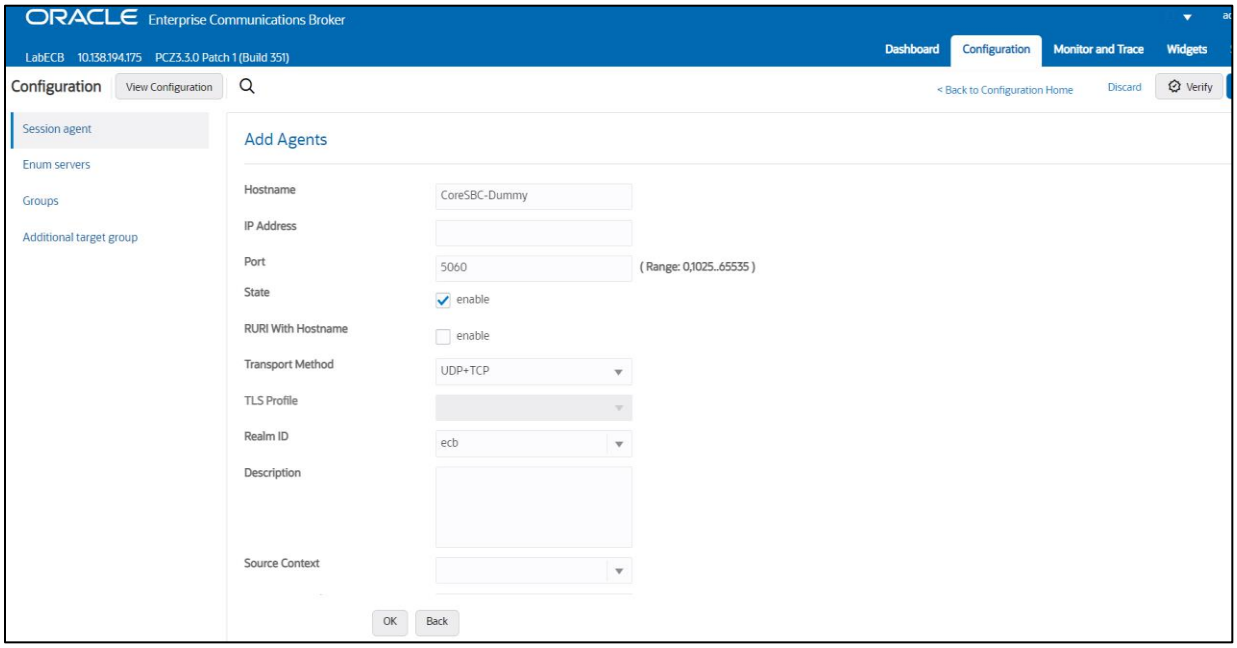
Please go to ECB and Navigate to Service Provisioning – Agents --- Additional Target Group --- Add,
To add a new ATG for parallel forking by giving Name – Any desired Name.

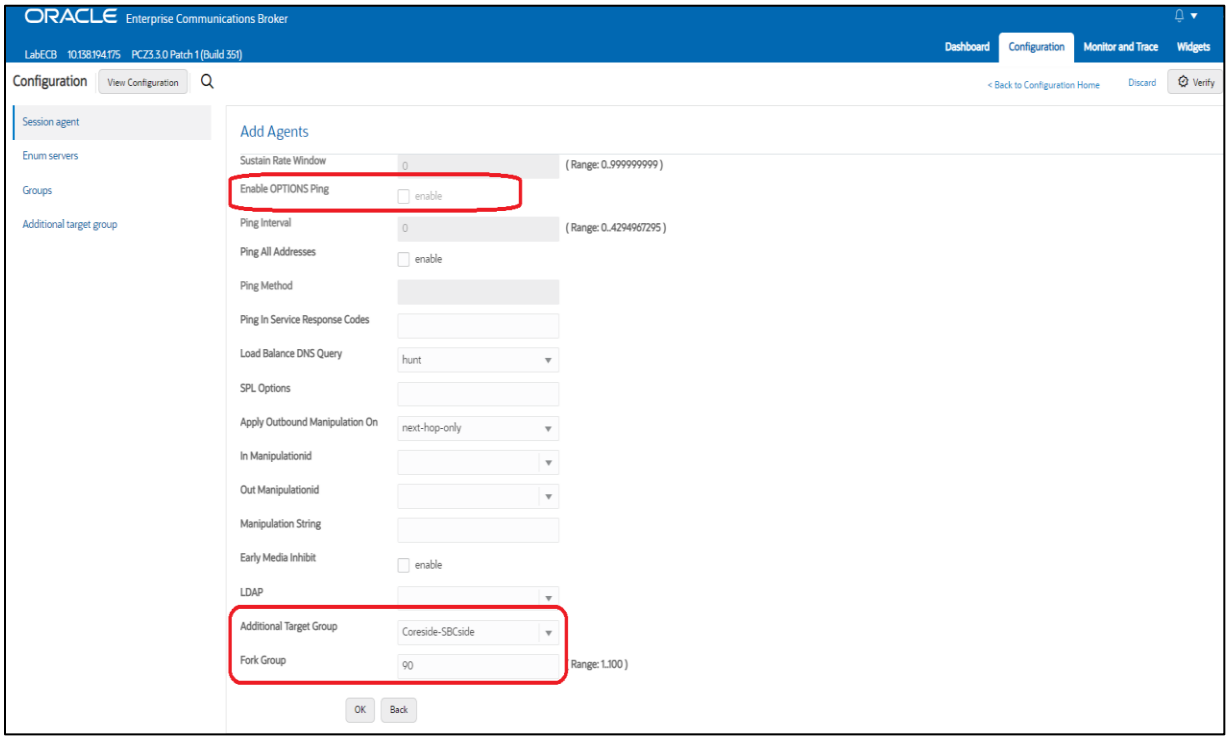
After that, select the Agents that you want to do parallel forking.
For this testing, we have selected Agents from both core side and SBC side.
Give Fork Group as equal for both Agents (10 in our case for both agents).



Go to: Service Provisioning ----- Agents ----- Session Agent.
Create a dummy agent with the following config for ATG parallel forking.

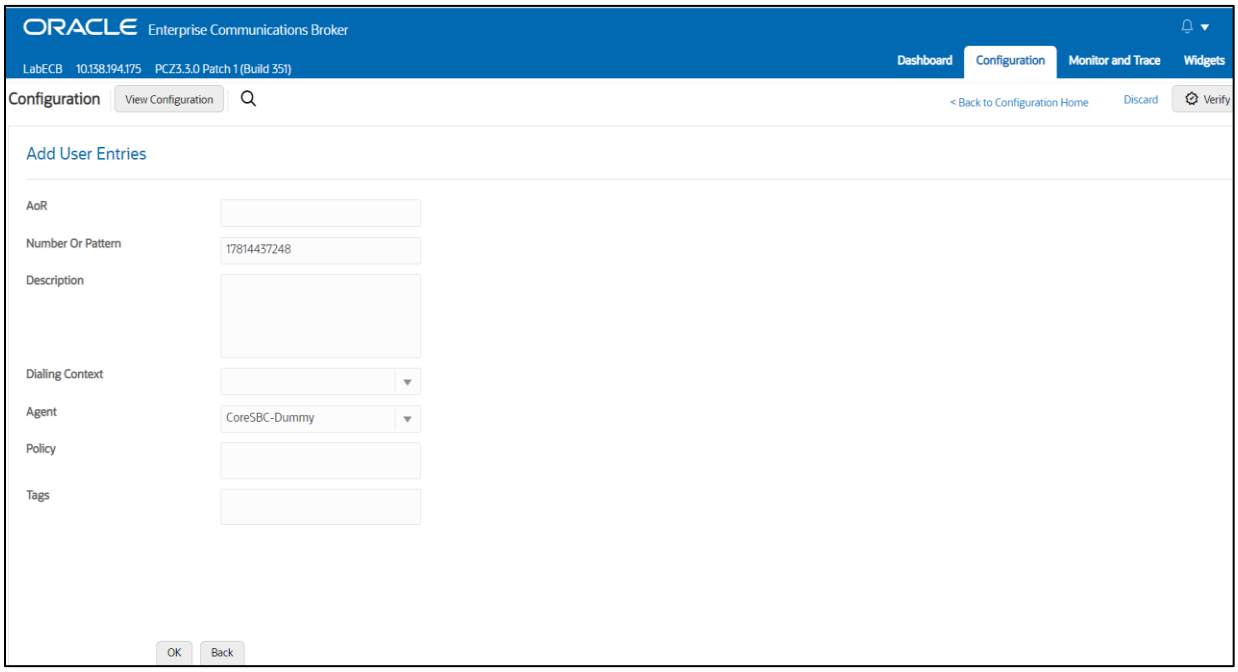
Hostname: Create a new agent (Dummy Agent) for the desired targets that require parallel forking.
IP Address: Leave blank for the Dummy Agent.
Enable OPTIONS ping: Should be disabled for dummy agent.
Additional target group: Assign the created ATG group
Fork group: The fork group number (90) must be higher than the values assigned to the Agents (10) within the ATG.





Go to: Service Provisioning ----- User Entries ----- Session Agent and Add a new entry as shown below.

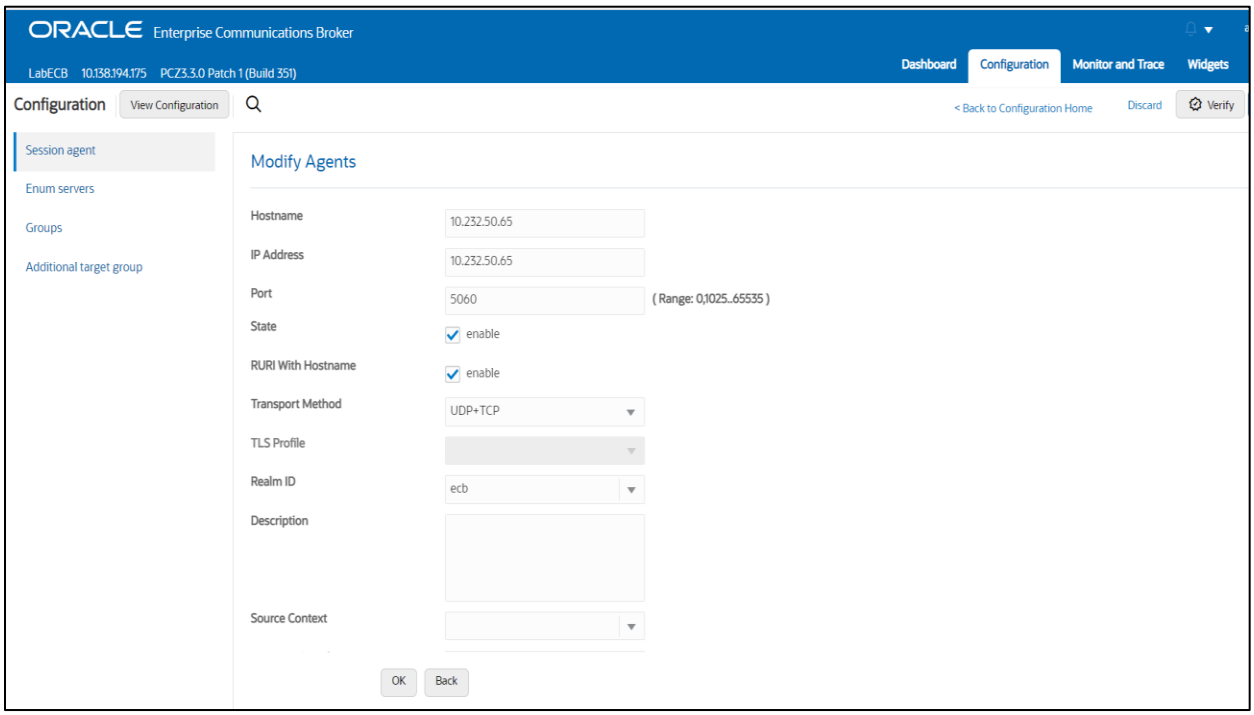
Number or Pattern: Enter the DID of the Agents
Agent: Select the Dummy Agent created.

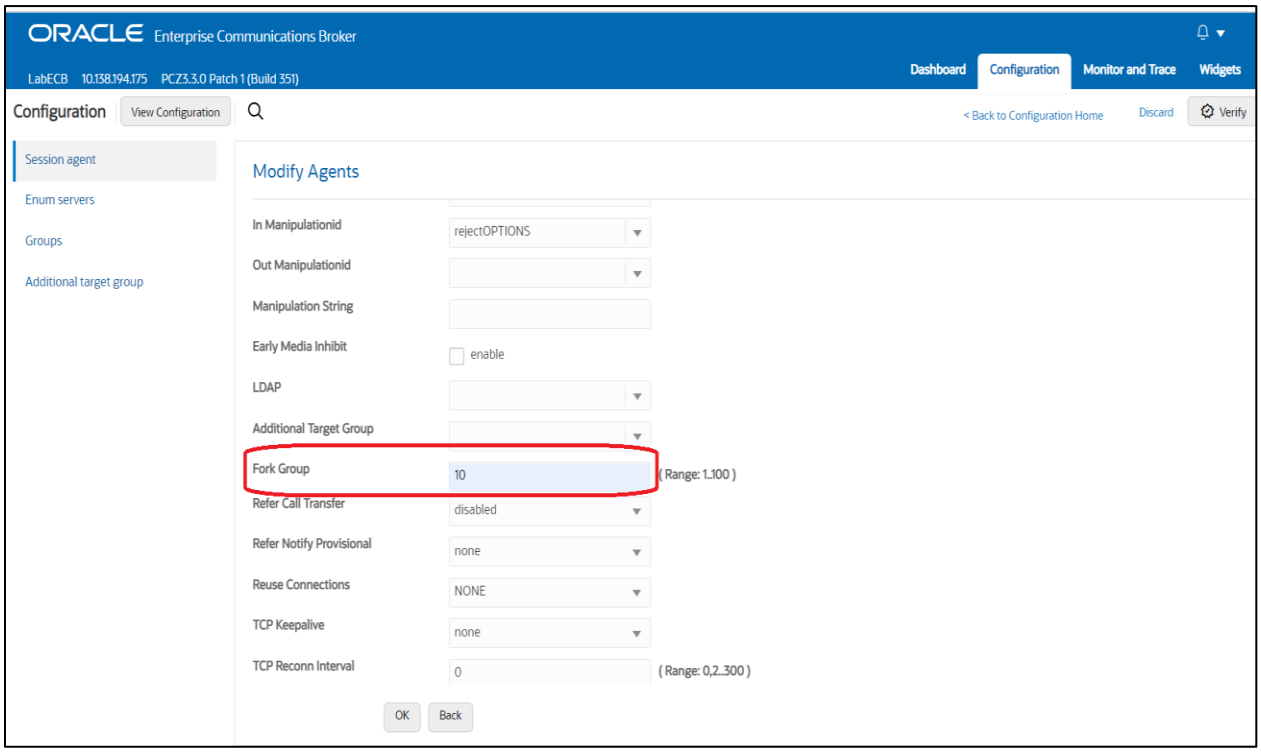


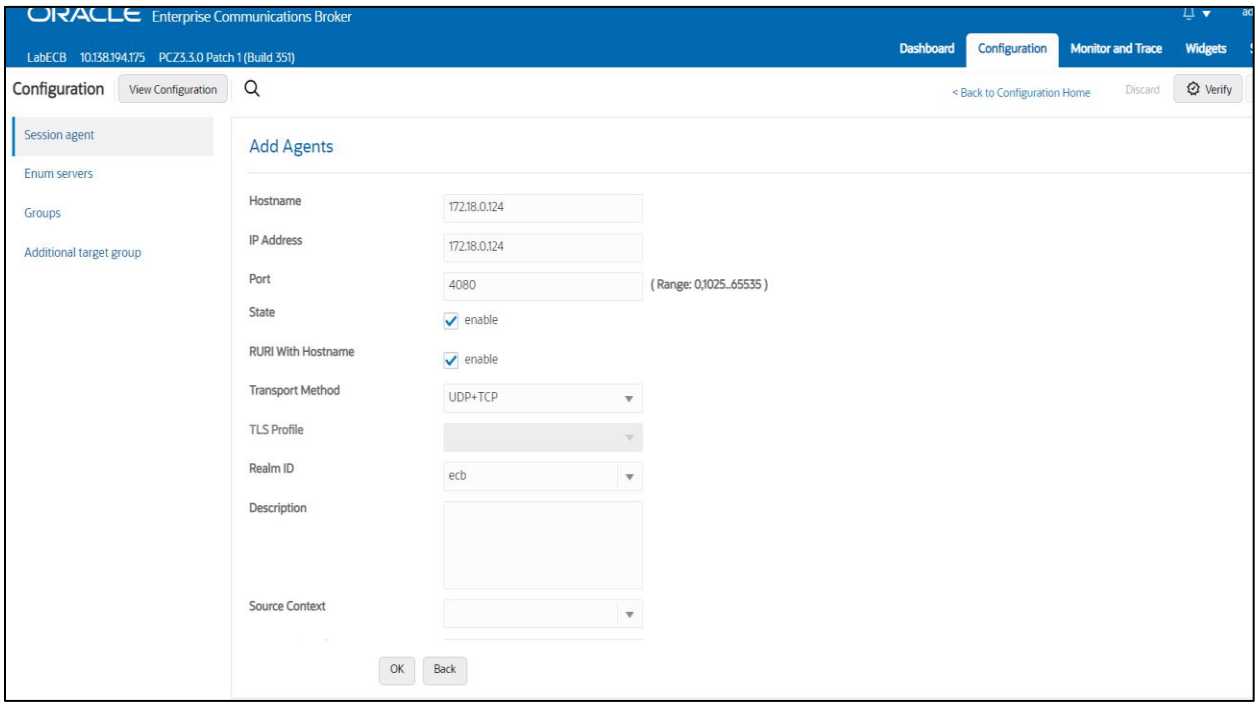
This DID will ring both Core side PBX (Genesys) and SBC side (Teams) due to parallel forking.

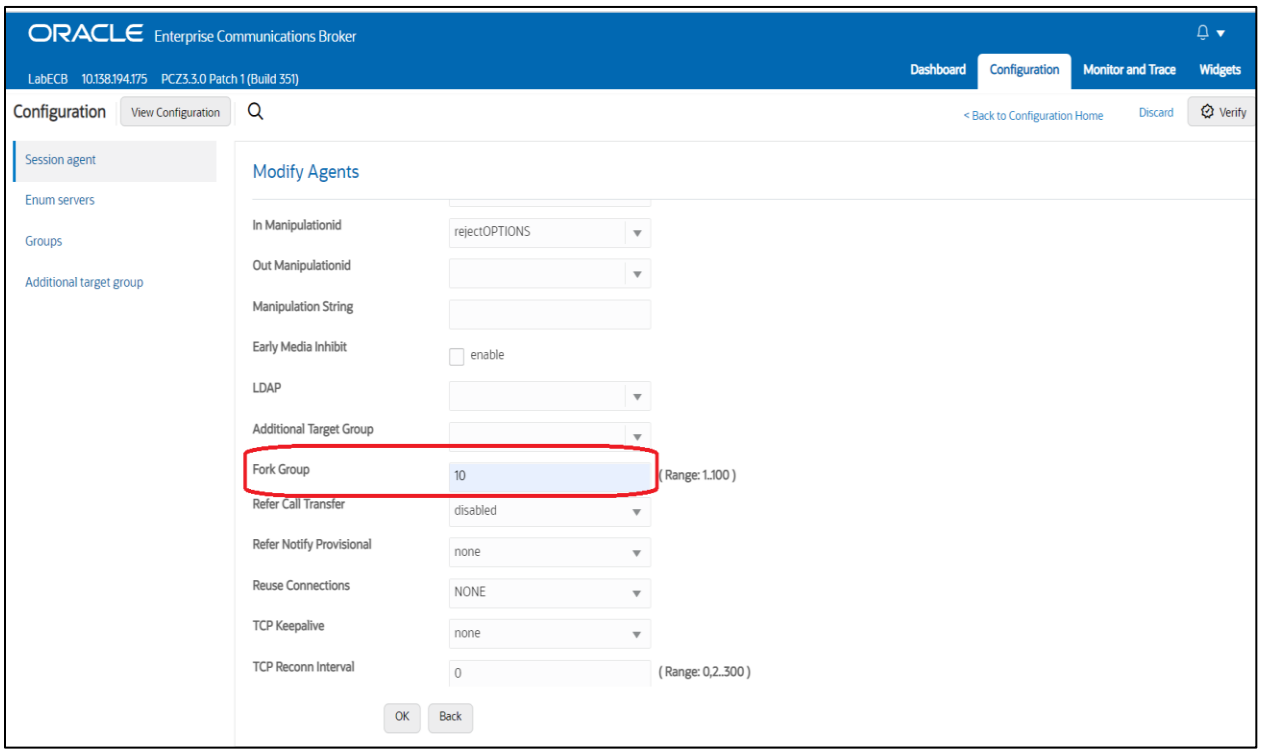
Finally, Go to: Service Provisioning ----- Agents ----- Session Agent.

The standard agent selected for Core and SBC side also need to be set to Fork group (10) to match the ATG fork group to make ATG parallel forking work properly. The config is shown below for reference. Save the config after that.









With this, ECB feature specific configuration are complete.

5. Configuring the SBC

This chapter strictly provides step-by-step guidance on how to configure Oracle SBC to route calls to Oracle ECB and ECB features related testing using Teams configuration as an example.

Note: This document will not cover SBC config for Teams and those config can be referred using any other Teams with SBC app notes.

5.1. Validated Oracle SBC version

Oracle conducted tests with Oracle SBC 8.4 software – this software with the configuration listed below can run on any of the following products:

- AP 1100
- AP 3900
- AP 4600
- AP 6350
- AP 6300
- VME

6. New SBC configuration

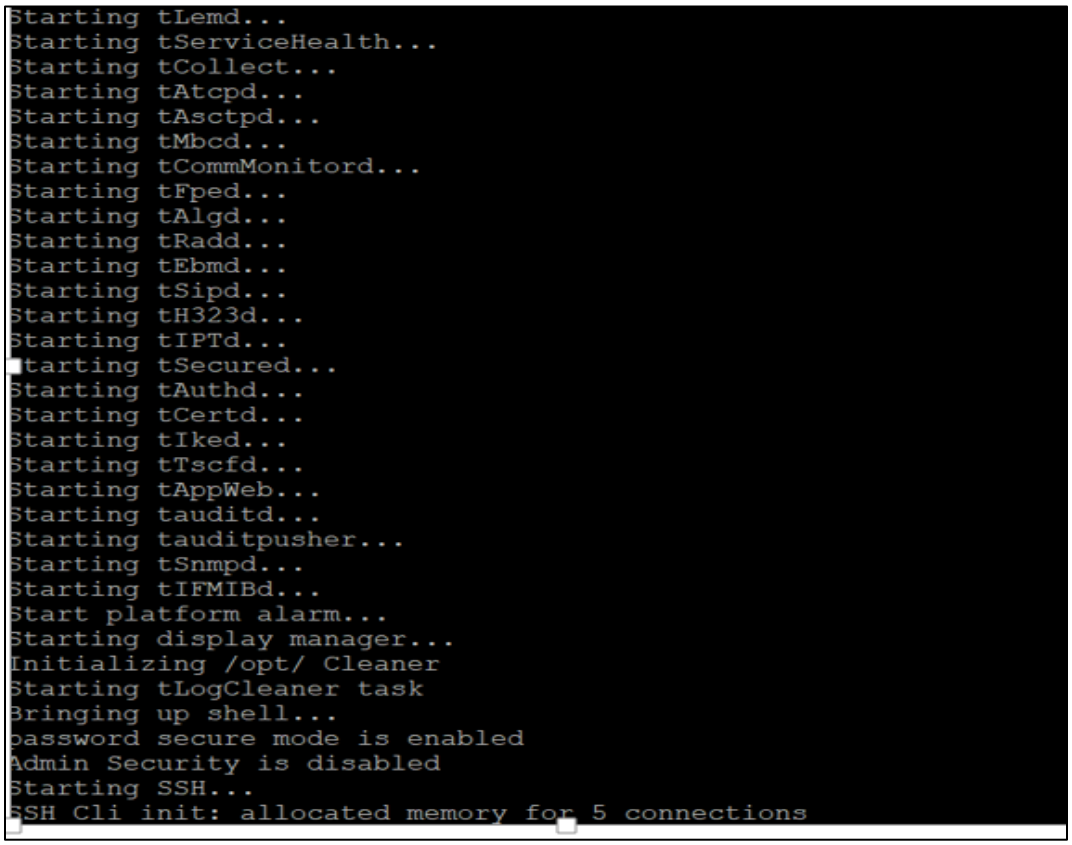
If the customer is looking to setup a new SBC from scratch, please follow the section below.

6.1. Establishing a serial connection to the SBC

Connect one end of a straight-through Ethernet cable to the front console port (which is active by default) on the SBC and the other end to console adapter that ships with the SBC, connect the console adapter (a DB-9 adapter) to the DB-9 port on a workstation, running a terminal emulator application such as Putty. Start the terminal emulation application using the following settings:

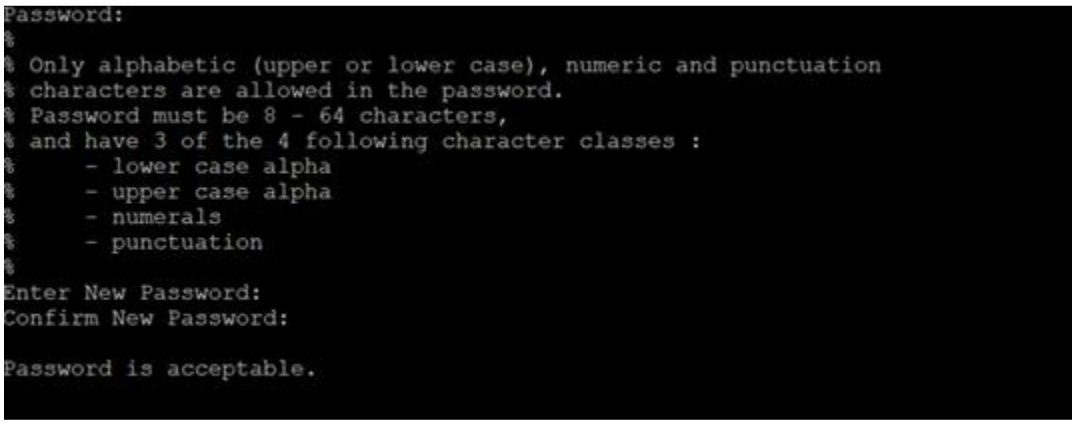
- Baud Rate=115200
- Data Bits=8
- Parity=None
- Stop Bits=1
- Flow Control=None

Power on the SBC and confirm that you see the following output from the boot-up sequence

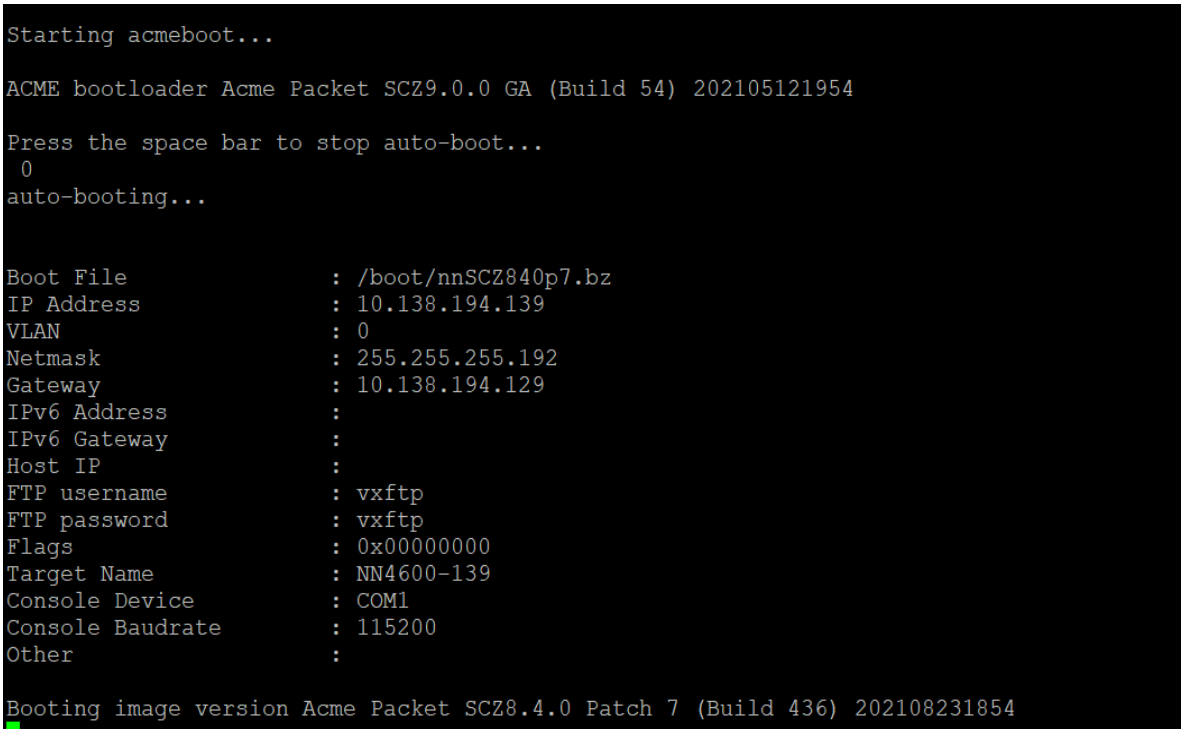


Enter the default password to log in to the SBC. Note that the default SBC password is “acme” and the default super user password is “packet”.

Both passwords have to be changed according to the rules shown below.



Now set the management IP of the SBC by setting the IP address in bootparam to access bootparam. Go to Configure terminal->bootparam.



Note: There is no management IP configured by default.

Setup product type to Enterprise Session Border Controller as shown below.

To configure product type, type in setup product in the terminal

```
NN4600-139#
NN4600-139# setup product

----------------------------------------------------------------
WARNING:
Alteration of product alone or in conjunction with entitlement
changes will not be complete until system reboot

Last Modified 2020-04-30 22:38:15
----------------------------------------------------------------
 1 : Product       : Enterprise Session Border Controller

Enter 1 to modify, d' to display, 's' to save, 'q' to exit. [s]:
```

Enable the features for the ESBC using the setup entitlements command as shown

Save the changes and reboot the SBC.

```
Entitlements for Enterprise Session Border Controller
Last Modified: Never
----------------------------------------------------------------
 1 : Session Capacity                        : 0
 2 :   Advanced                              :
 3 : Admin Security                          :
 4 : Data Integrity (FIPS 140-2)             :
 5 : Transcode Codec AMR Capacity            : 0
 6 : Transcode Codec AMRWB Capacity          : 0
 7 : Transcode Codec EVRC Capacity           : 0
 8 : Transcode Codec EVRCB Capacity          : 0
 9 : Transcode Codec EVS Capacity            : 0
 10: Transcode Codec OPUS Capacity           : 0
 11: Transcode Codec SILK Capacity           : 0

Enter 1 - 11 to modify, d' to display, 's' to save, 'q' to exit. [s]: 1

  Session Capacity (0-128000)                : 500

Enter 1 - 11 to modify, d' to display, 's' to save, 'q' to exit. [s]: 3

************************************************************
CAUTION: Enabling this feature activates enhanced security
functions. Once saved, security cannot be reverted without
resetting the system back to factory default state.
************************************************************
  Admin Security (enabled/disabled)          :

Enter 1 - 11 to modify, d' to display, 's' to save, 'q' to exit. [s]: 5

  Transcode Codec AMR Capacity (0-102375)    : 50

Enter 1 - 11 to modify, d' to display, 's' to save, 'q' to exit. [s]: 2

    Advanced (enabled/disabled)              : enabled

Enter 1 - 11 to modify, d' to display, 's' to save, 'q' to exit. [s]: 10

  Transcode Codec OPUS Capacity (0-102375)   : 50

Enter 1 - 11 to modify, d' to display, 's' to save, 'q' to exit. [s]: 11

  Transcode Codec SILK Capacity (0-102375)   : 50
```

The SBC comes up after reboot and is now ready for configuration.

Go to configure terminal->system->http-server-config.

Enable the http-server-config to access the SBC using Web GUI. Save and activate the config.

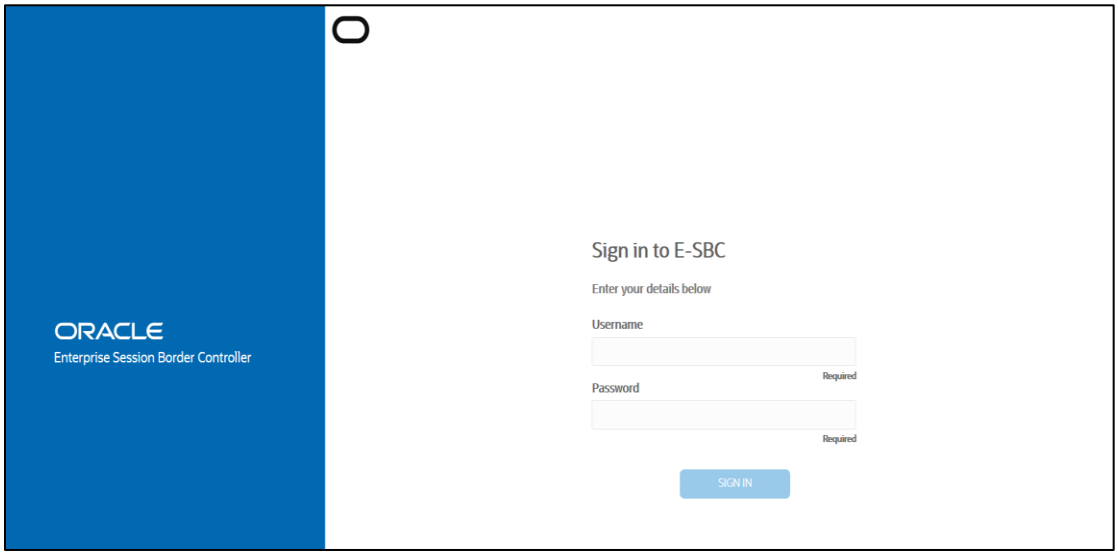
```
NN4600-139(http-server)#
NN4600-139(http-server)# show
http-server
        name                                    webServerInstance
        state                                   enabled
        realm
        ip-address
        http-state                              enabled
        http-port                               80
        https-state                             disabled
        https-port                              443
        http-interface-list                     REST,GUI
        http-file-upload-size                   0
        tls-profile
        auth-profile
        last-modified-by                        @
        last-modified-date                      2021-01-25 00:16:28

NN4600-139(http-server)#
```

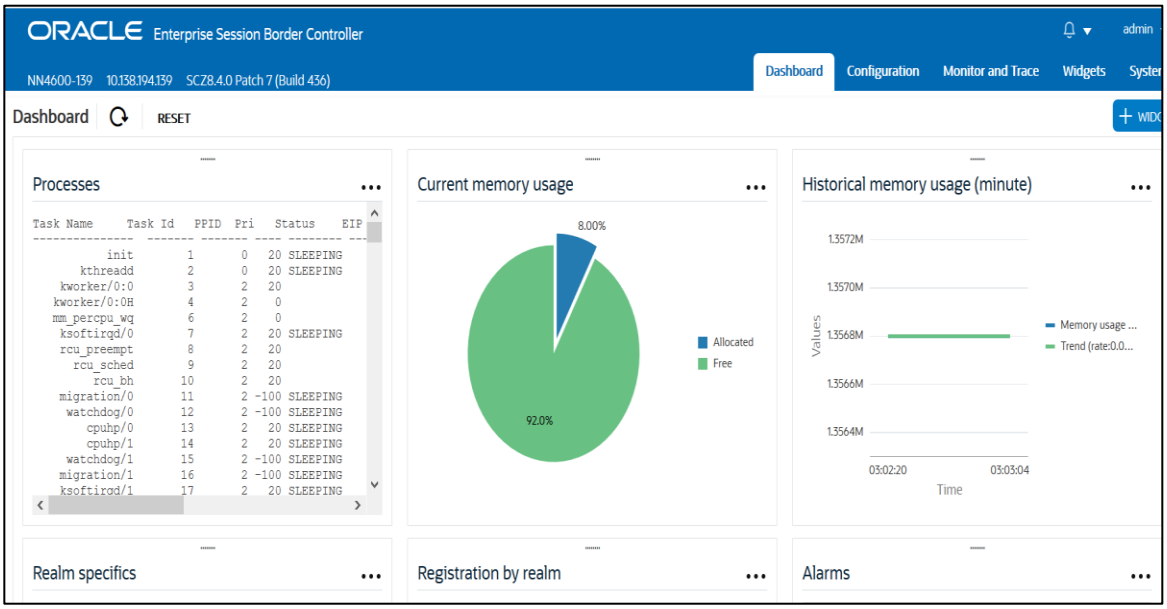
6.2. Configure SBC using Web GUI

In this app note, we configure SBC using the WebGUI.

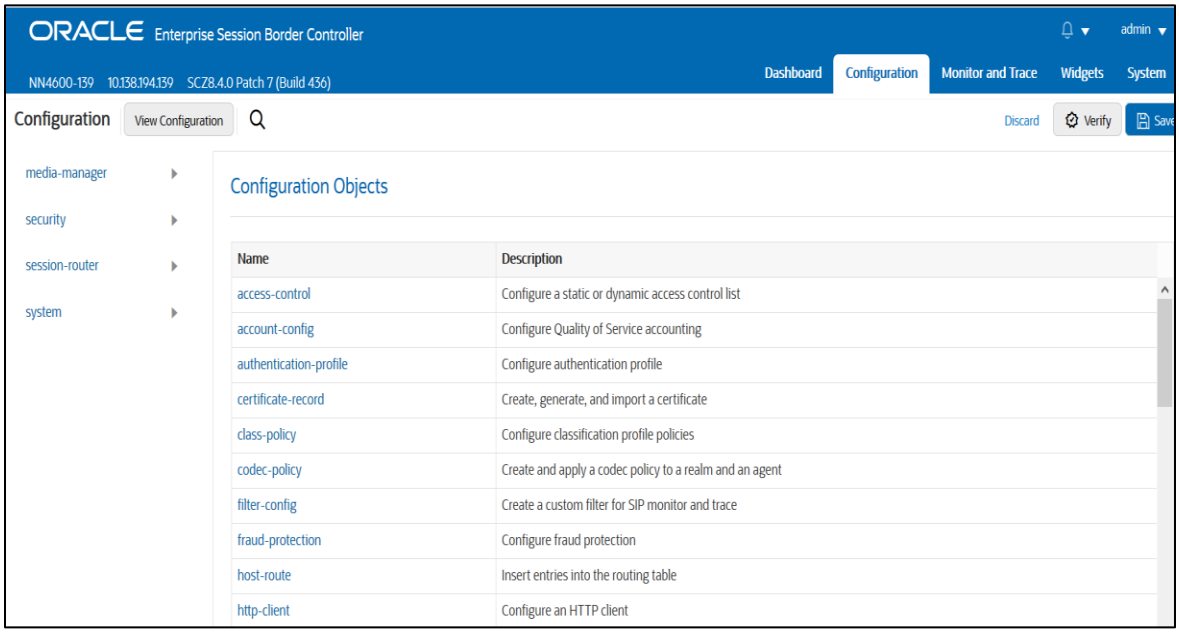
The Web GUI can be accessed through the url http://<SBC_MGMT_IP>.



The username and password is the same as that of CLI.



Go to Configuration as shown below, to configure the SBC



Kindly refer to the GUI User Guide given below for more information.

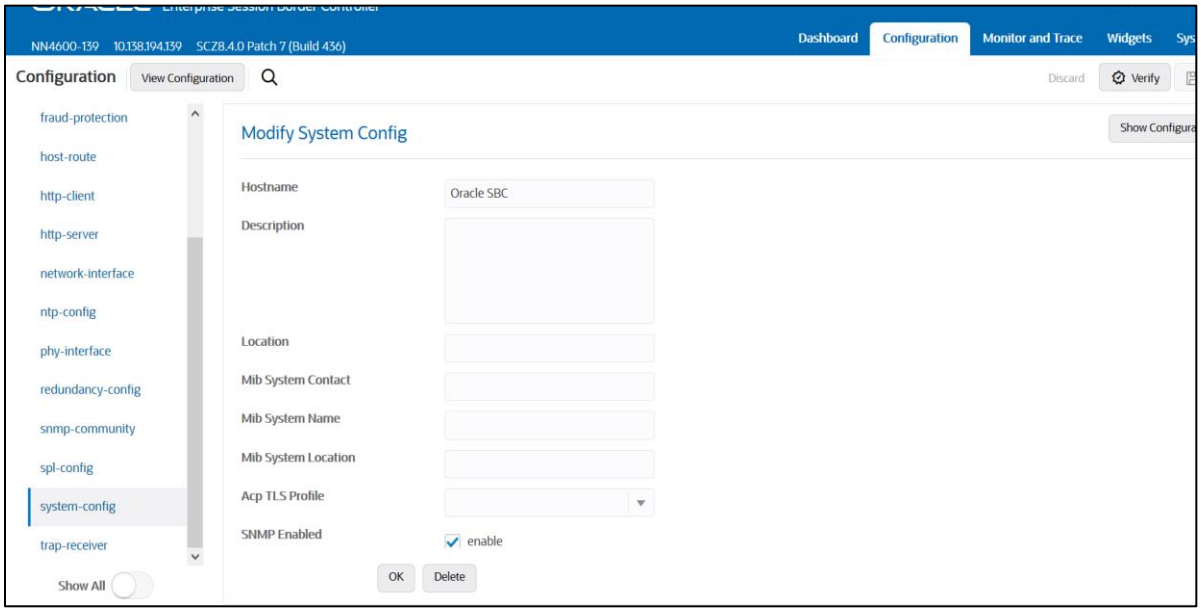
https://docs.oracle.com/en/industries/communications/enterprise-session-border-controller/8.4.0/webgui/esbc_scz840_webgui.pdf

The expert mode is used for configuration.

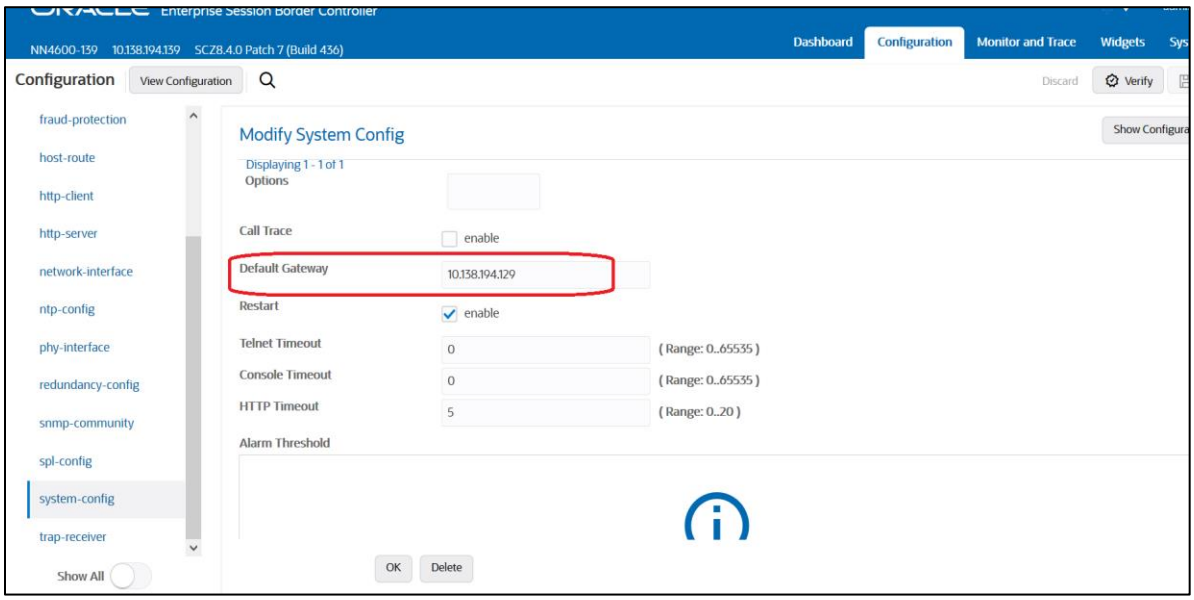
Tip: To make this configuration simpler, one can directly search the element to be configured, from the Objects tab available.

6.3. Configure system-config

Go to system->system-config



Please enter the default gateway value in the system config page.



For VME, transcoding cores are required. Please refer the documentation here for more information

https://docs.oracle.com/en/industries/communications/enterprise-session-border-controller/8.4.0/releasenotes/esbc_scz840_releasenotes.pdf

The above step is needed only if any transcoding is used in the configuration.
If there is no transcoding involved, then the above step is not needed.

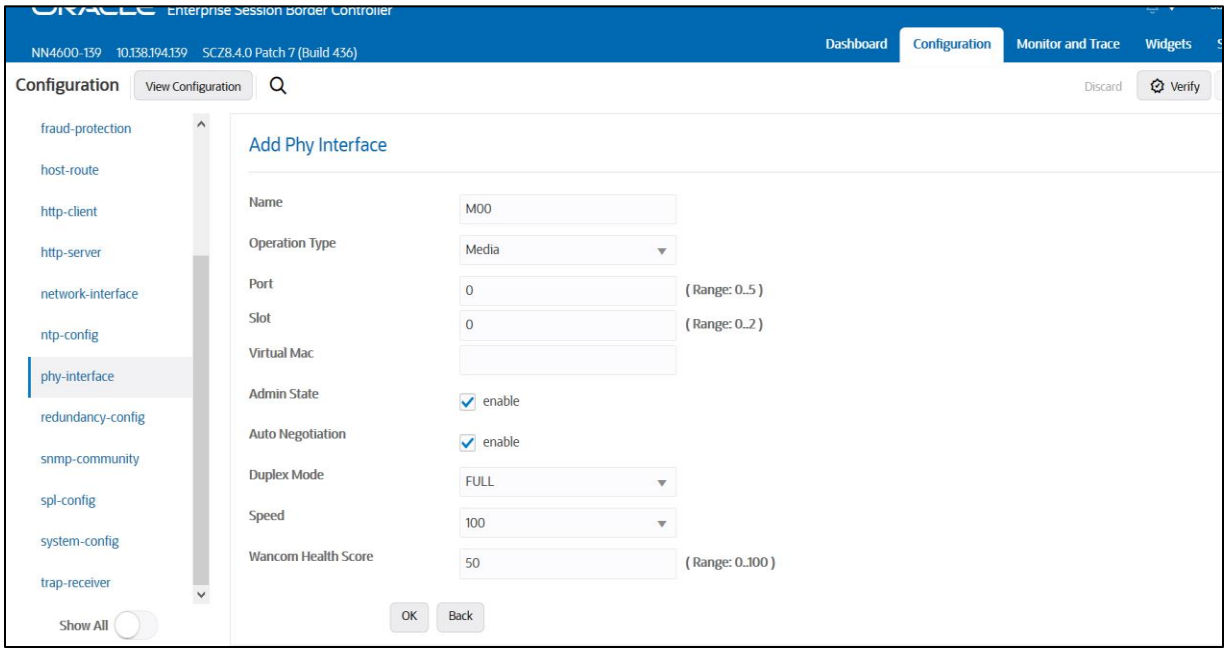
6.4. Configure Physical Interface values

To configure physical Interface values, go to System->phy-interface.

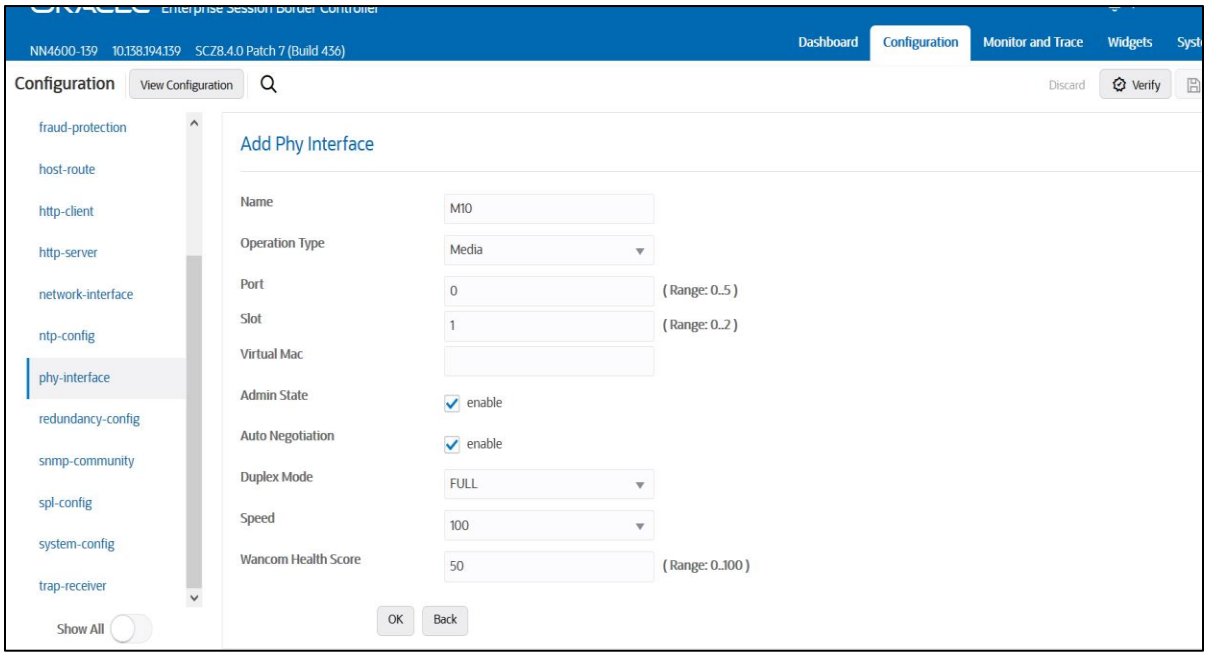
You will first configure the slot 0, port 0 interface designated with the name M00.
This will be the port plugged into your public interface. (For Teams and Verizon side)
Avaya side is configured on the slot 0 port 1

Parameter Name	Public Interface(M00)	ECB Side (M10)
Slot	0	0
Port	0	1
Operation Mode	Media	Media

Please configure M00 interface as below.



Similarly, configure M10 interface as below.



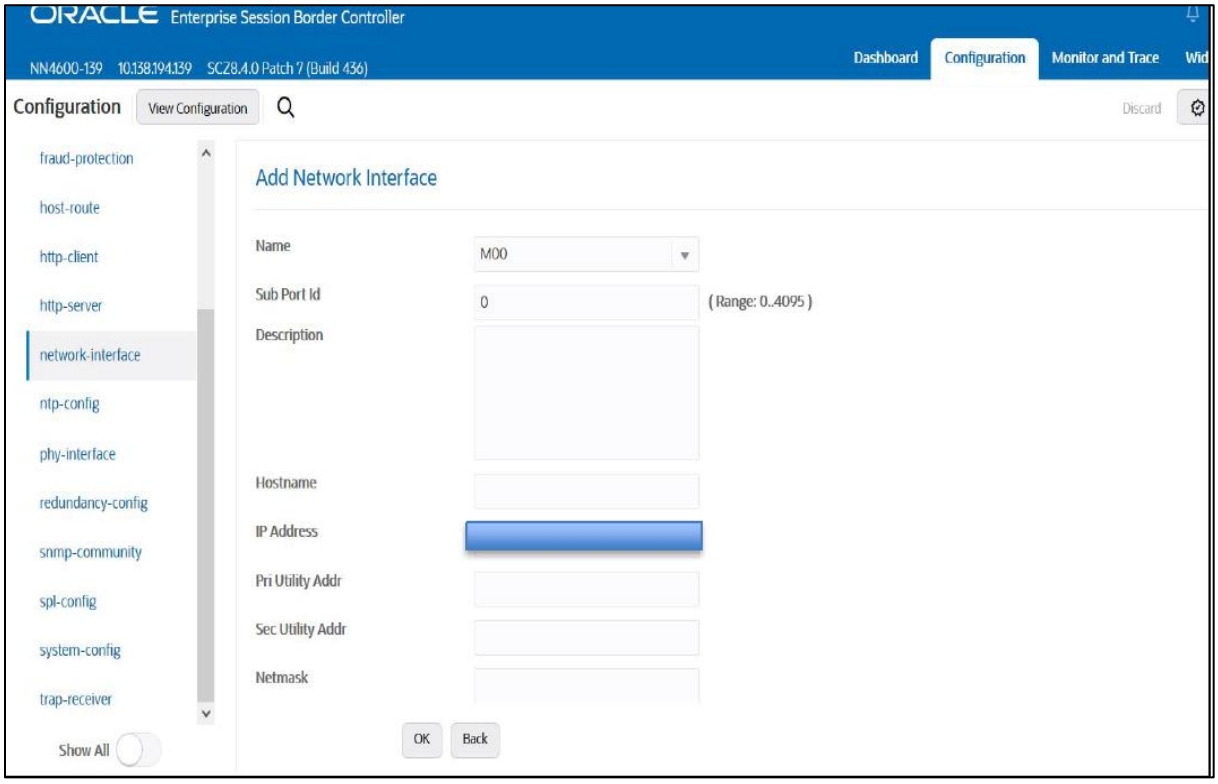
6.5. Configure Network Interface values

To configure network-interface, go to system->Network-Interface. Configure two interfaces

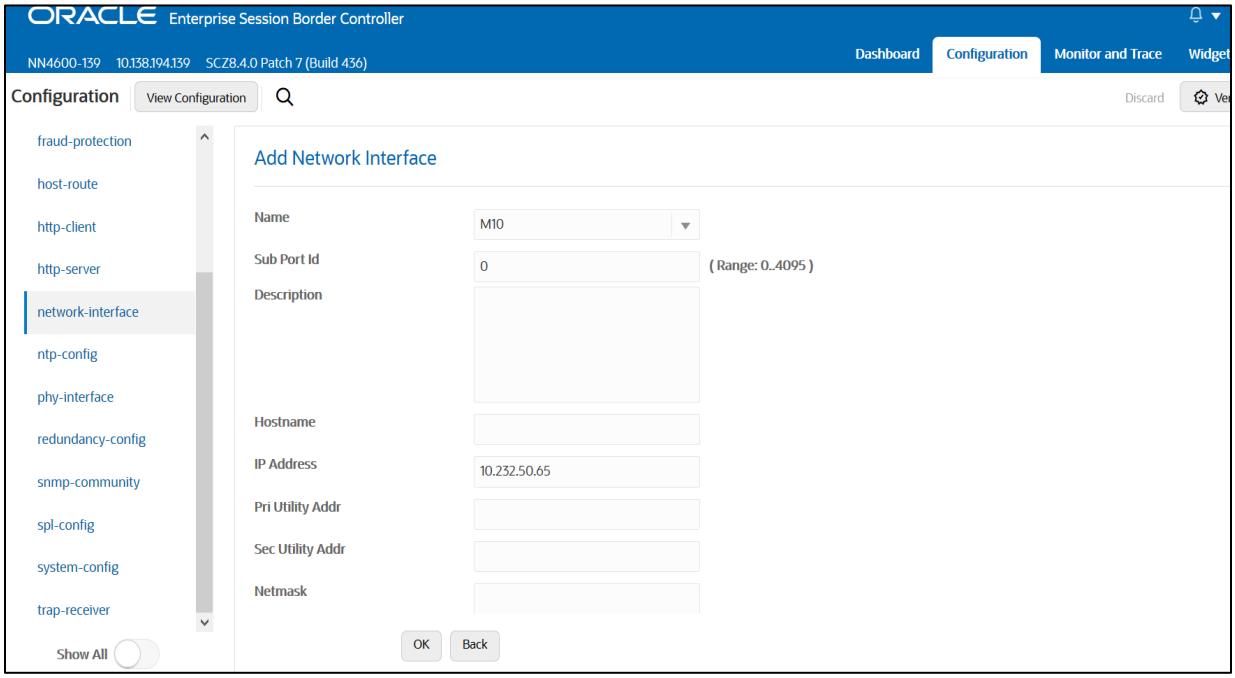
The table below lists the parameters, to be configured for both the interfaces.

Parameter Name	Public Interface (For Teams)	ECB side Network interface
Name	M00	M10
Host Name	customers.telechat.o-test06161977.com	
IP address	[redacted]	10.232.50.65
Netmask	255.255.255.192	255.255.255.0
Gateway	[redacted]	10.232.50.1

Please configure network interface M00 as below



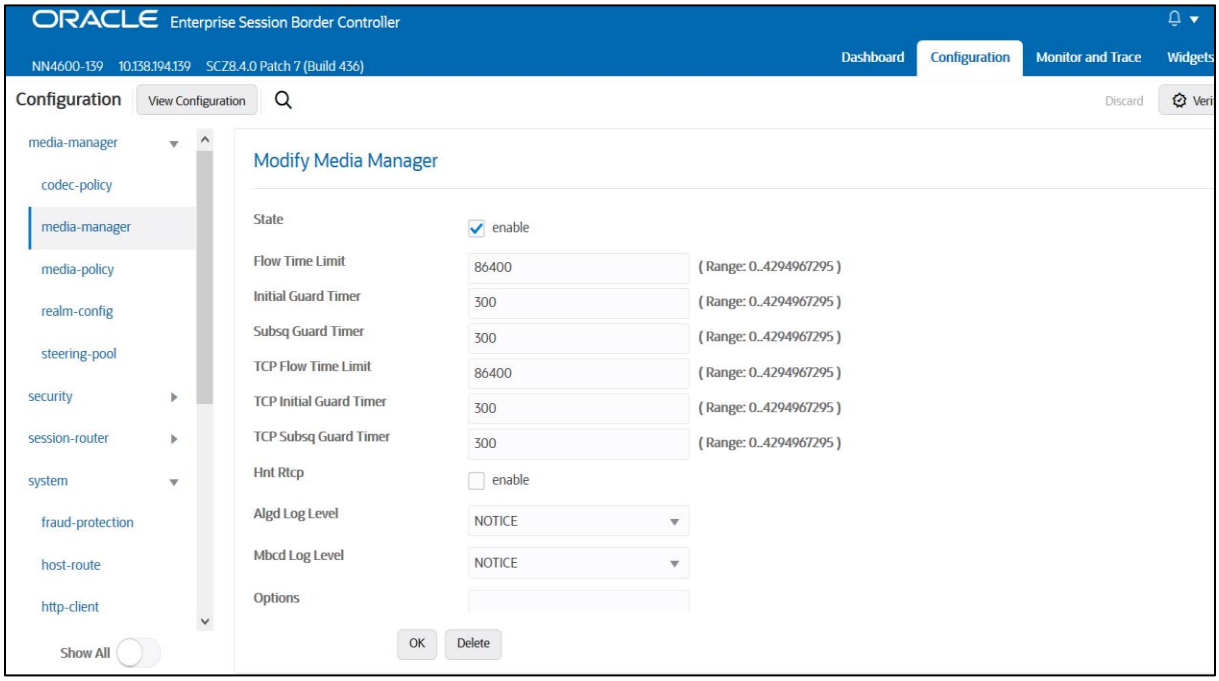
Please configure network interface M10 as below

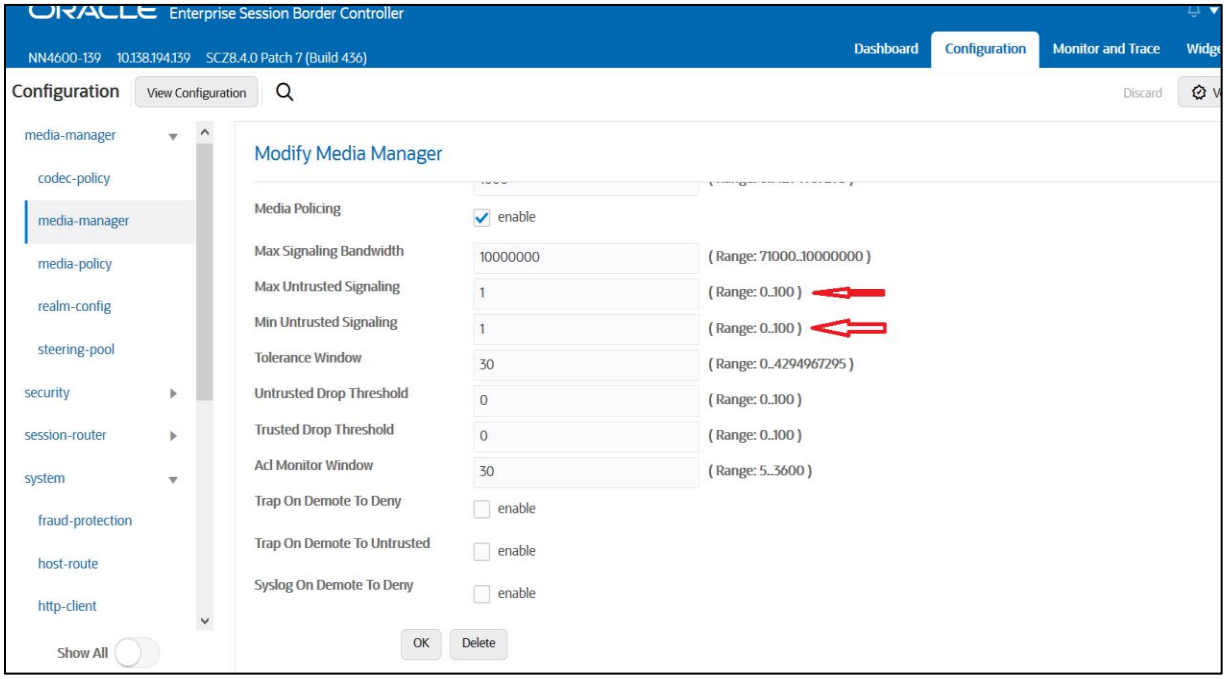


6.6. Enable media manager

Media-manager handles the media stack required for SIP sessions on the SBC. Enable the media manager option as below.

In addition to the above config, please set the max and min untrusted signaling values to 1.
Go to Media-Manager->Media-Manager





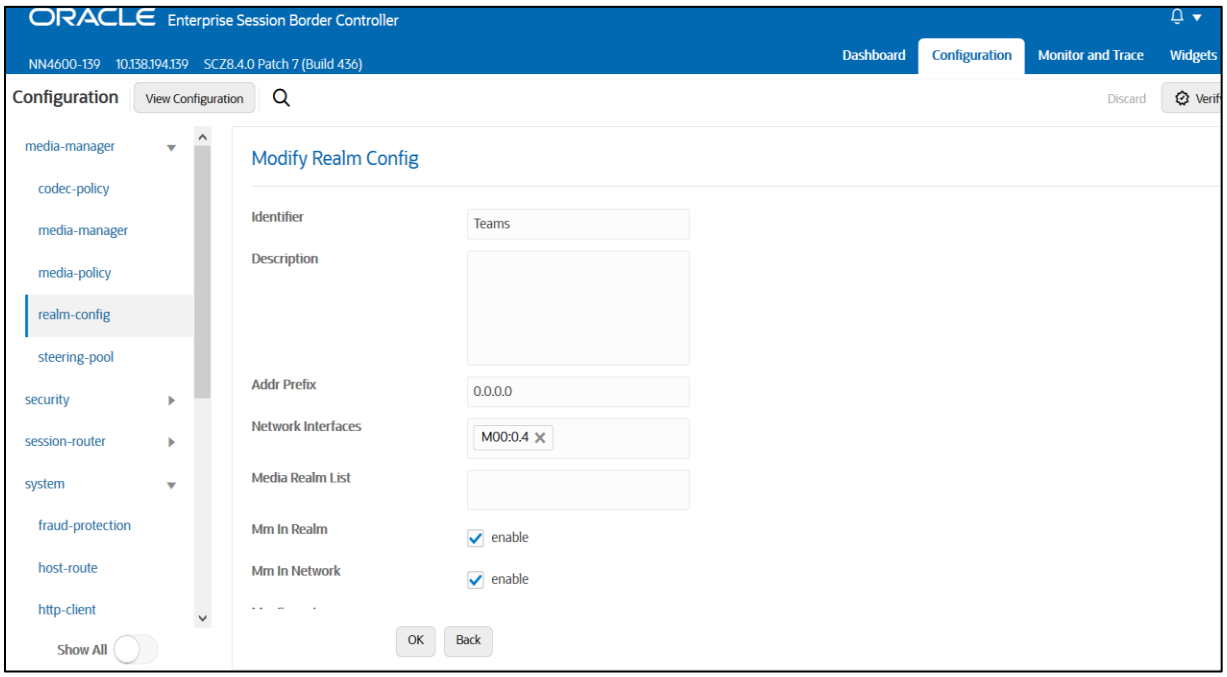
6.7. Configure Realms

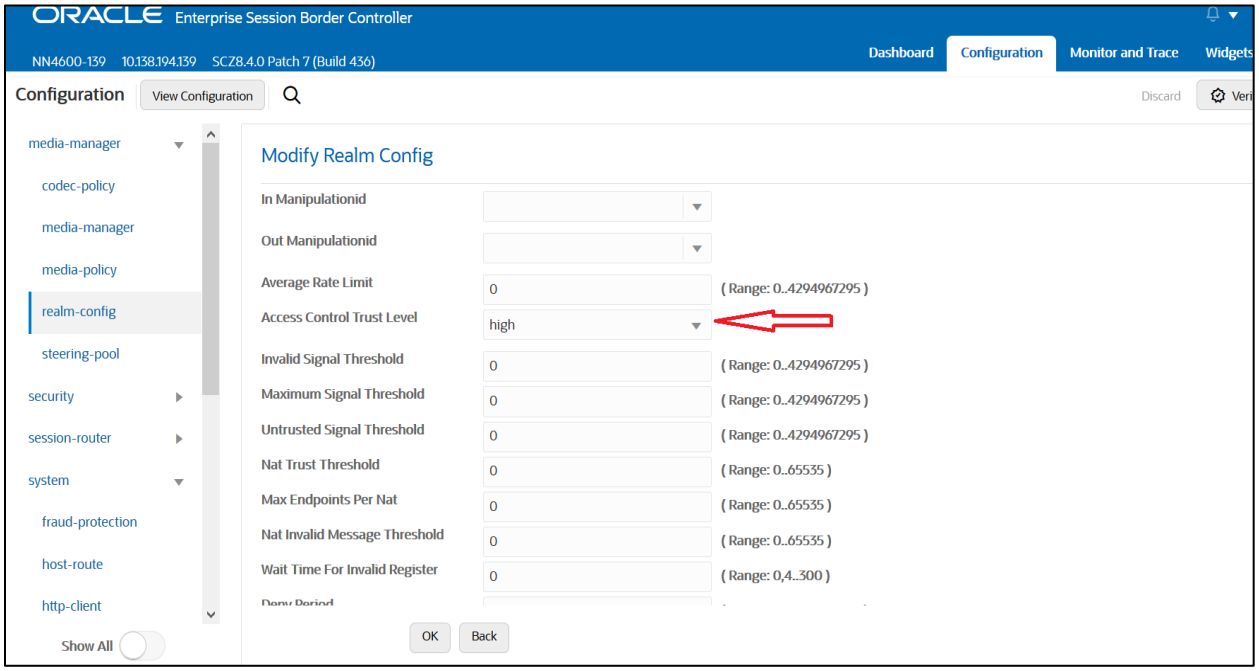
Navigate to realm-config under media-manager and configure a realm as shown below
The name of the Realm can be any relevant name according to the user convenience.

Use the following table as a configuration example for the three realms used in this configuration:

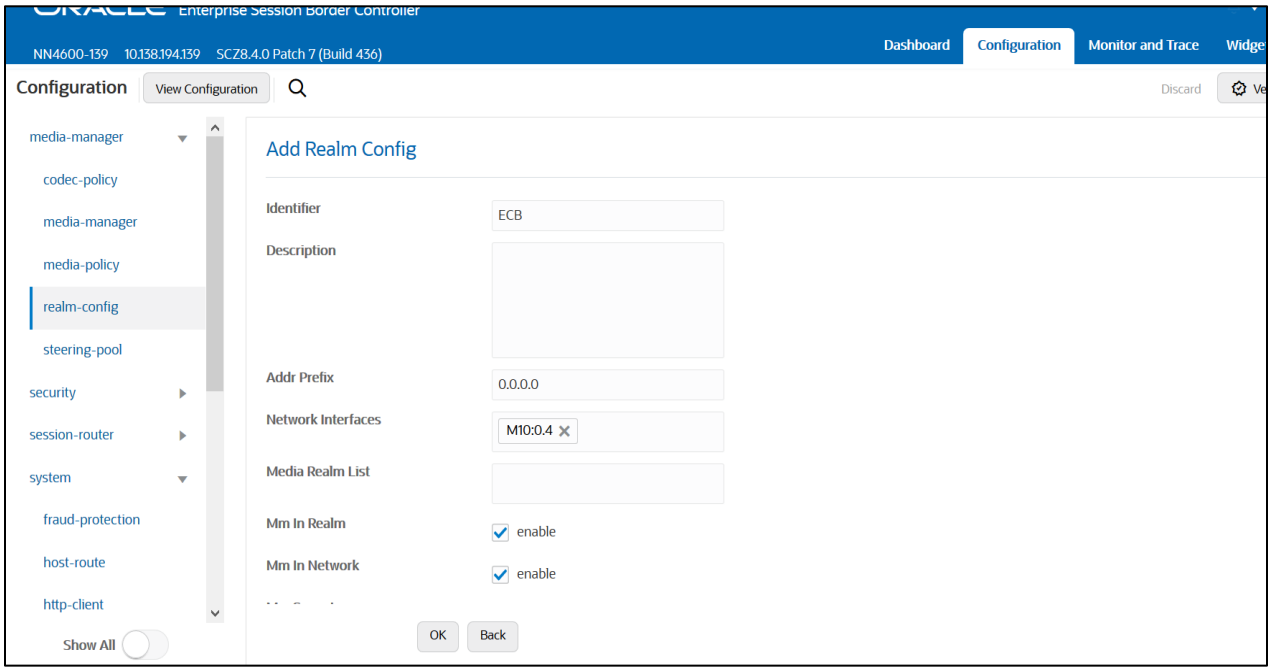
Config Parameter	Teams Realm	ECB Realm
Identifier	Teams	Avaya Realm
Network Interface	M00	M10
Mm in realm	☑	☑
Teams-FQDN	Telechat.o-test06161977.com	
Teams fqdn in uri	☑	
Sdp inactive only	☑	
Media Sec policy	sdespolicy	RTP
RTCP mux	☑	
ice profile	ice	
Codec policy	addCN	OptimizeCodecs
RTCP policy	rtcpGen	
Access Control Trust Level	High	High
Pai-strip	Enabled	enabled
Refer Call Transfer	Enabled	

In the below case, Realm name is given as Teams for Teams Side.
Please set the Access Control Trust Level as high for this realm





Finally, Realm name is given as ECB for ECB Side.
Please set the Access Control Trust Level to high for this realm



For more information on Access Control Trust Level, please refer to SBC Security guide link given below:

https://docs.oracle.com/en/industries/communications/session-border-controller/8.4.0/security/sbc_scz840_security.pdf

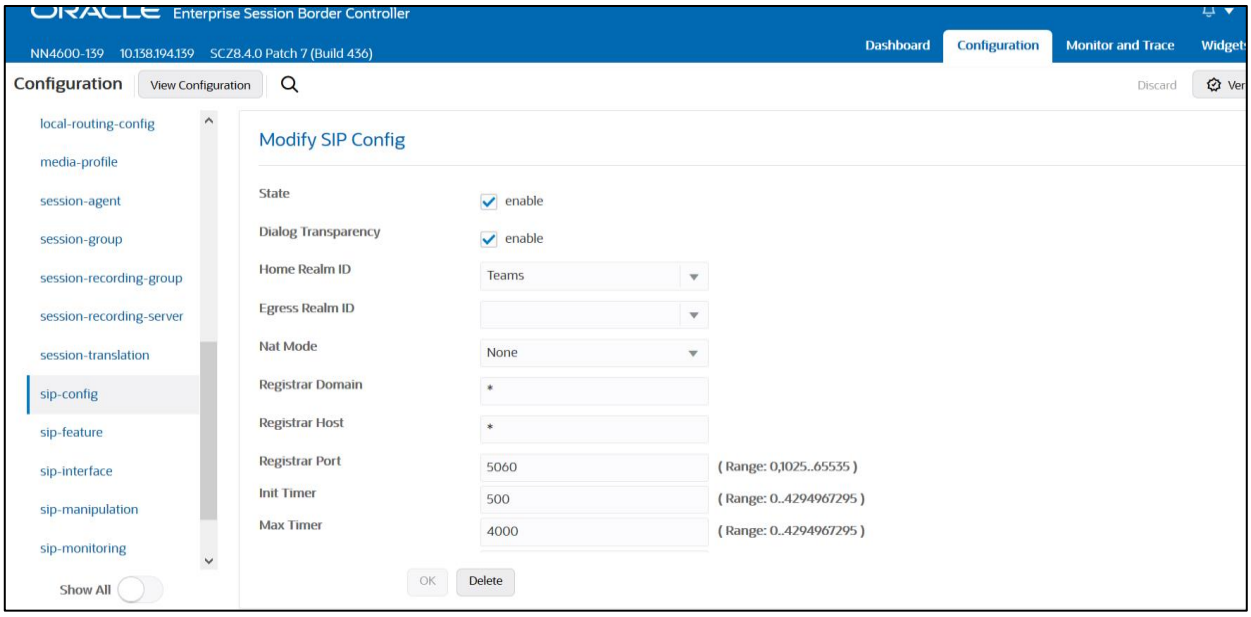
6.8. Enable sip-config

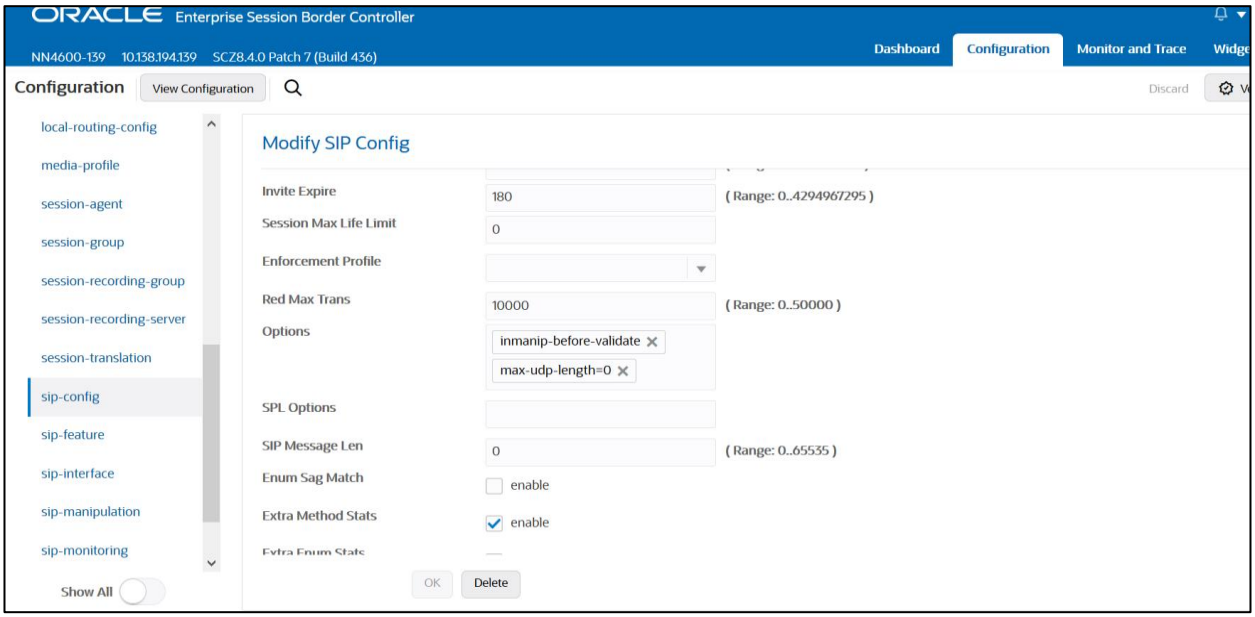
SIP config enables SIP handling in the SBC.
Make sure the home realm-id, registrar-domain and registrar-host are configured.

Also add the options to the sip-config as shown below.
To configure sip-config, Go to Session-Router->sip-config and in options, add the below

- add max-udp-length =0
- inmanip-before-validate

For more info, please refer to SBC security guide given in the above section.



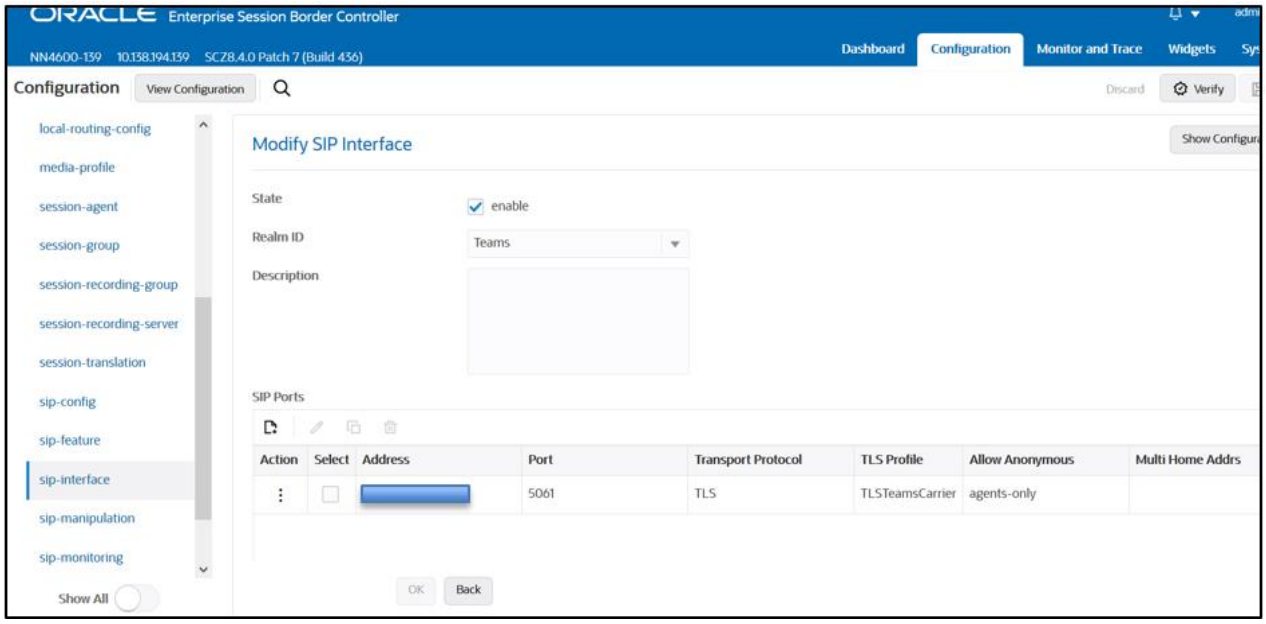


6.9. Configure SIP Interfaces

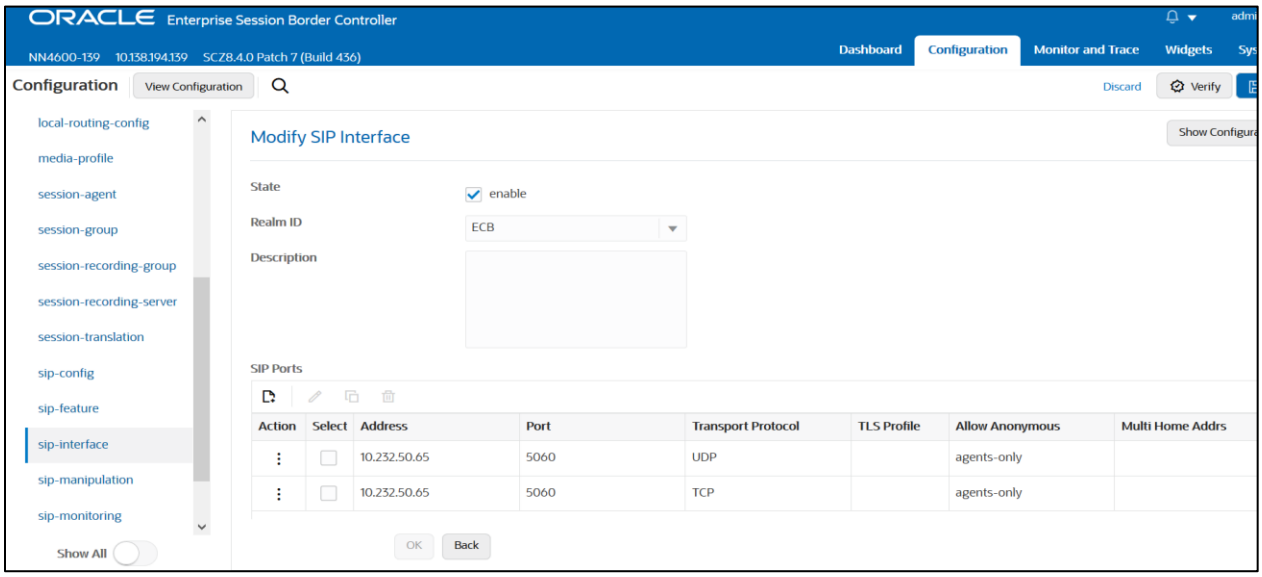
Navigate to sip-interface under session-router and configure the sip-interface as shown below. Please configure the below settings under the sip-interface.

- Tls-profile needs to match the name of the tls-profile created (Not covered in this app note)
- Set allow-anonymous to agents-only to ensure traffic to this sip-interface only comes from the particular Session agents added to the SBC.

Below is the sip-interface Configured for Teams side.



Similarly, configure sip-interface for ECB side as below



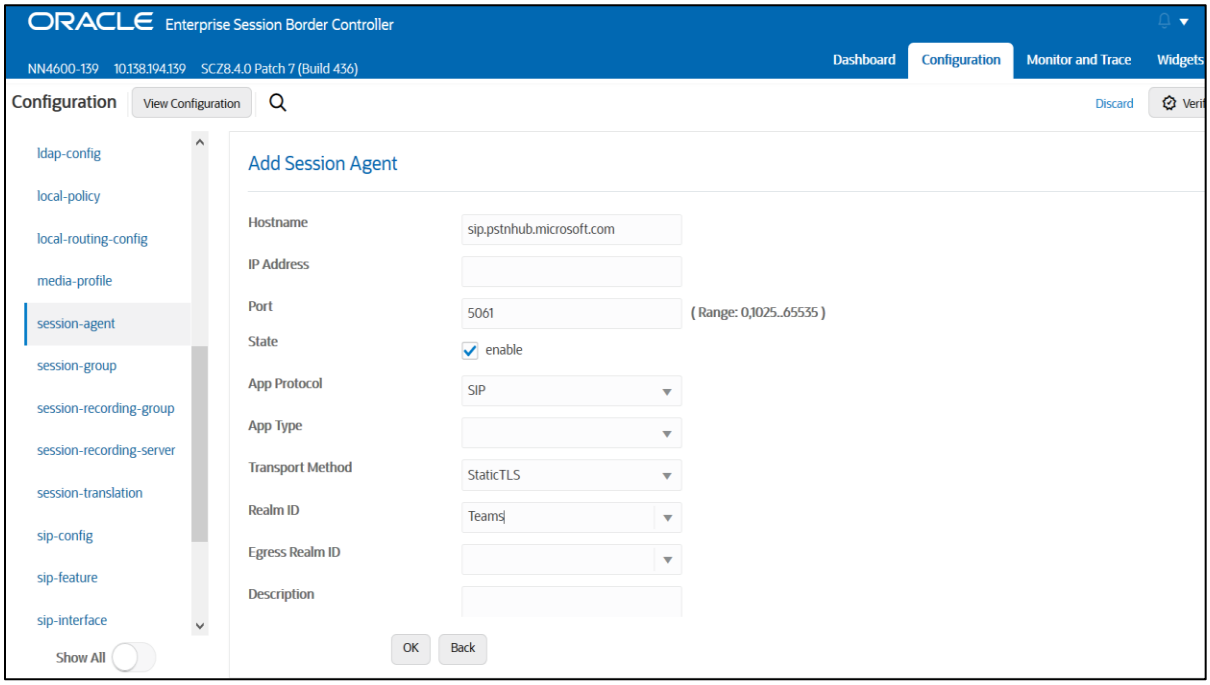
Once sip-interface is configured – the SBC is ready to accept traffic on the allocated IP address.

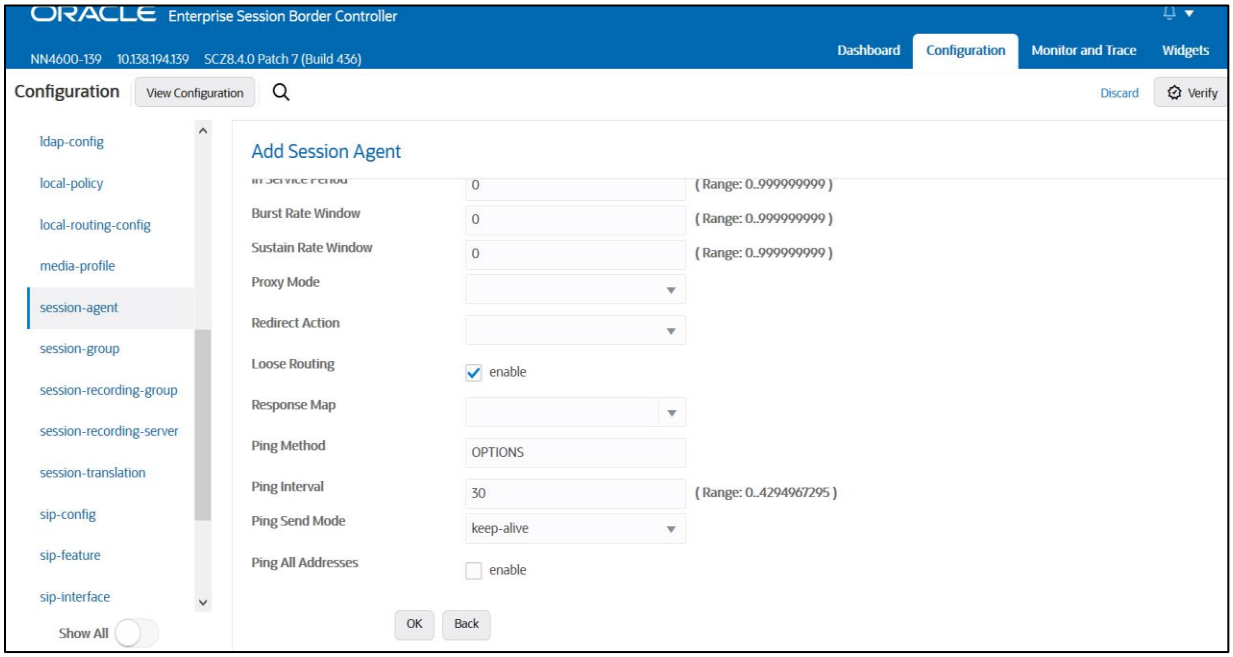
6.10. Configure session-agent

Session-agents are config elements which are trusted agents who can send/receive traffic from the SBC with direct access to trusted data path. Session-agents are config elements which are trusted agents who can send/receive traffic from the SBC with direct access to trusted data path.

Configure the session-agent for Teams with the following parameters.
Go to session-router->Session-Agent.

- hostname to “sip.pstnhub.microsoft.com”
- port 5061
- realm-id – needs to match the realm created for Teams
- transport set to “StaticTLS”
- refer-call-transfer set to enabled
- ping-method – send OPTIONS message to Microsoft to check health
- ping-interval to 30 secs
- Refer Call Transfer set to Enabled





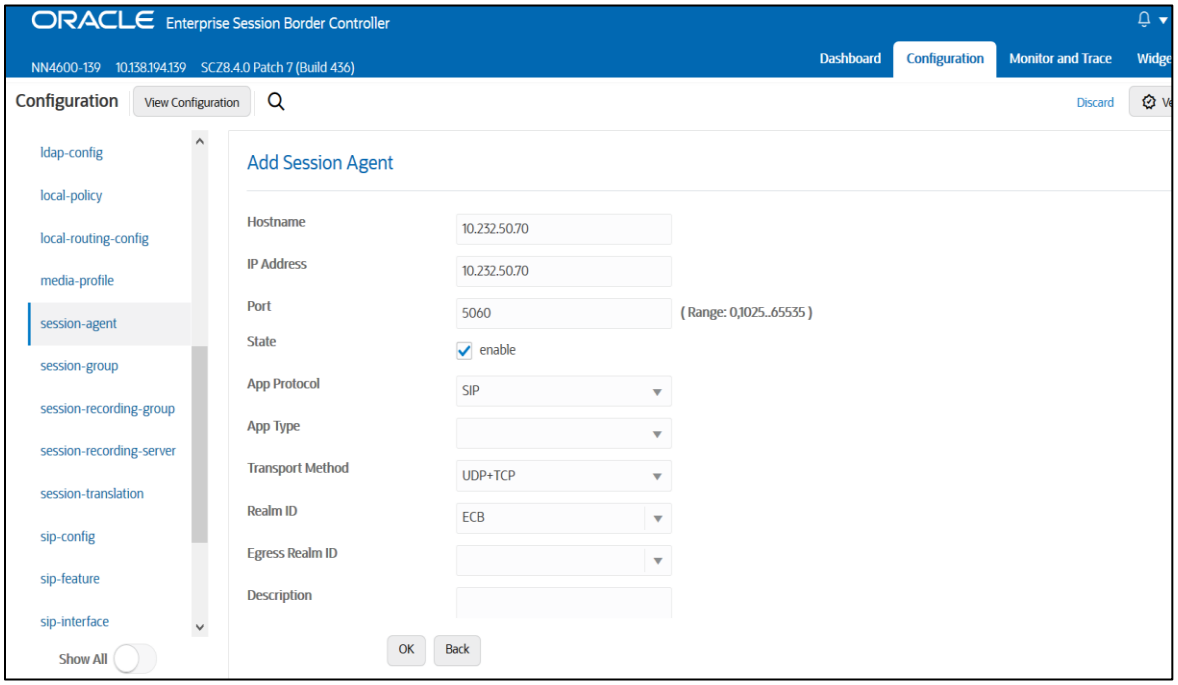
Follow above steps to create 2 more sessions for:

- sip2.pstnhub.microsoft.com
- sip3.pstnhub.microsoft.com

We also need to create Session Agent Group and sip-manipulations for Teams side which is not covered in this document.

Finally, Configure the session-agent for ECB Side which is Oracle ECB where SBC should route the calls. Go to session-router->Session-Agent.

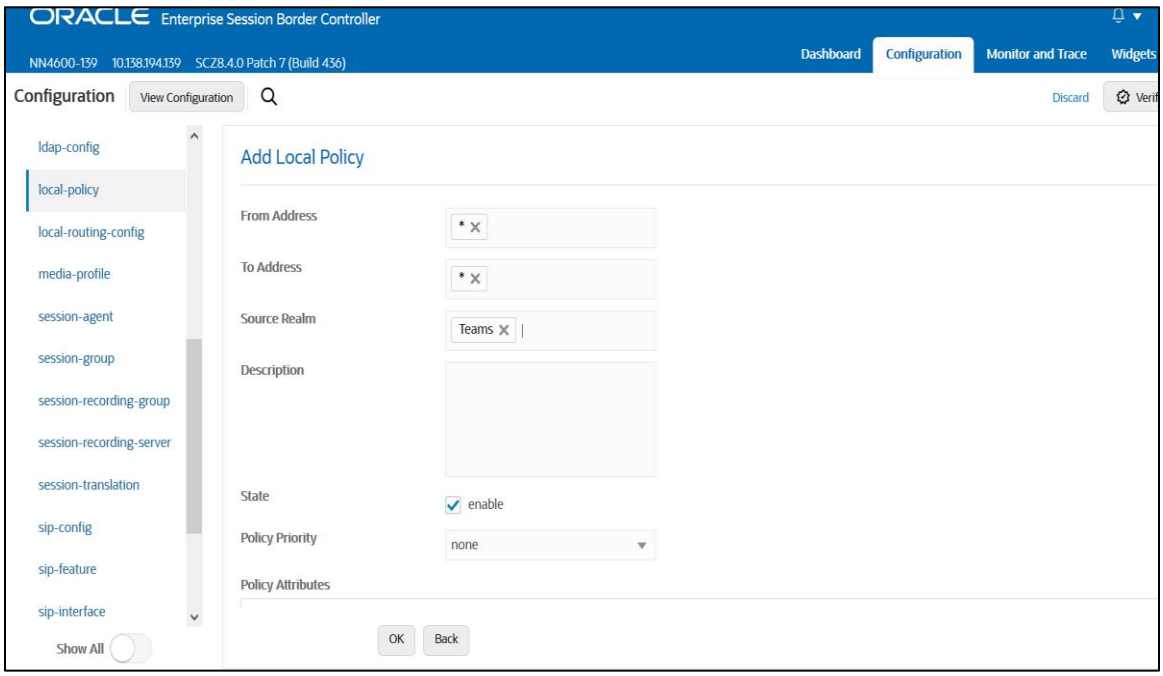
- **Host name and IP address to 10.232.50.70 which is the ECB IP**.
- port 5060
- realm-id – needs to match the realm created for ECB Side.
- transport set to “UDP+TCP

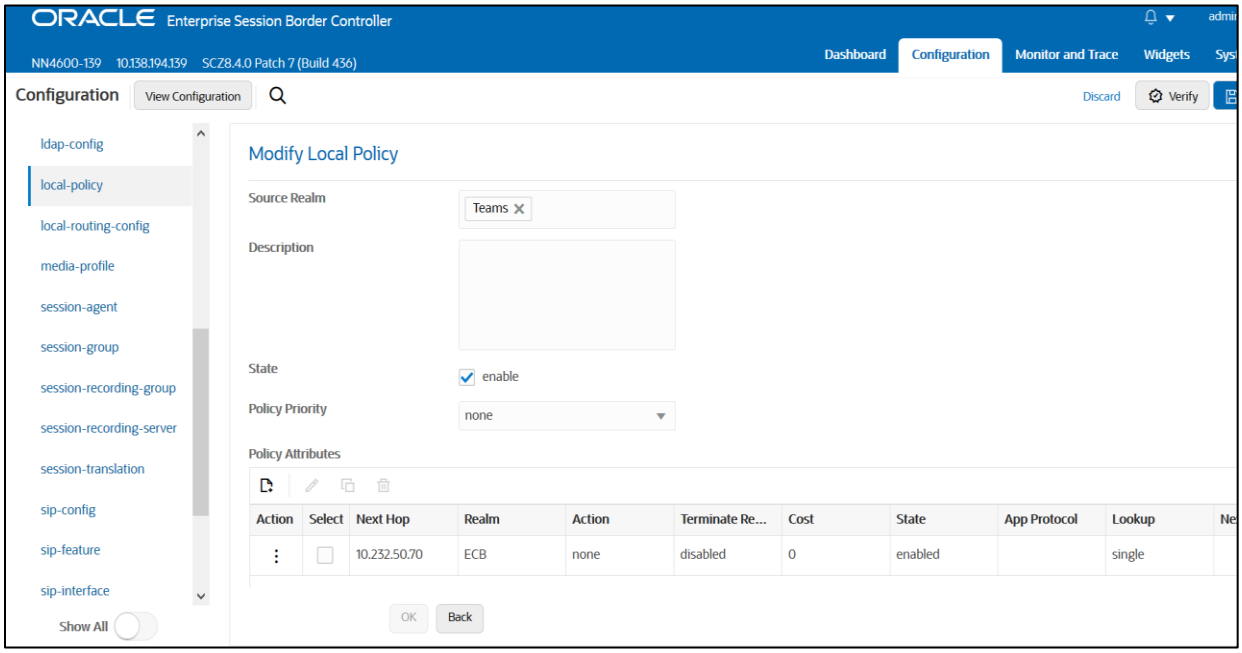


6.11. Configure local-policy

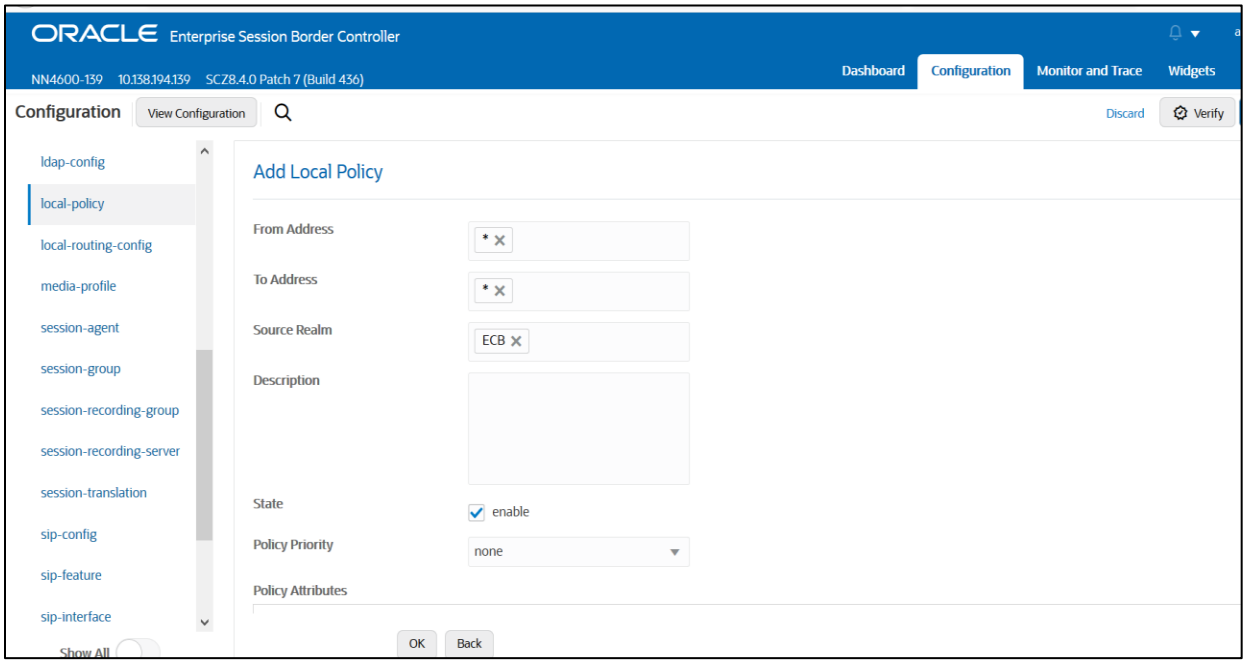
Local policy config allows for the SBC to route calls from one end of the network to the other based on routing criteria. To configure local-policy, go to Session-Router->local-policy.

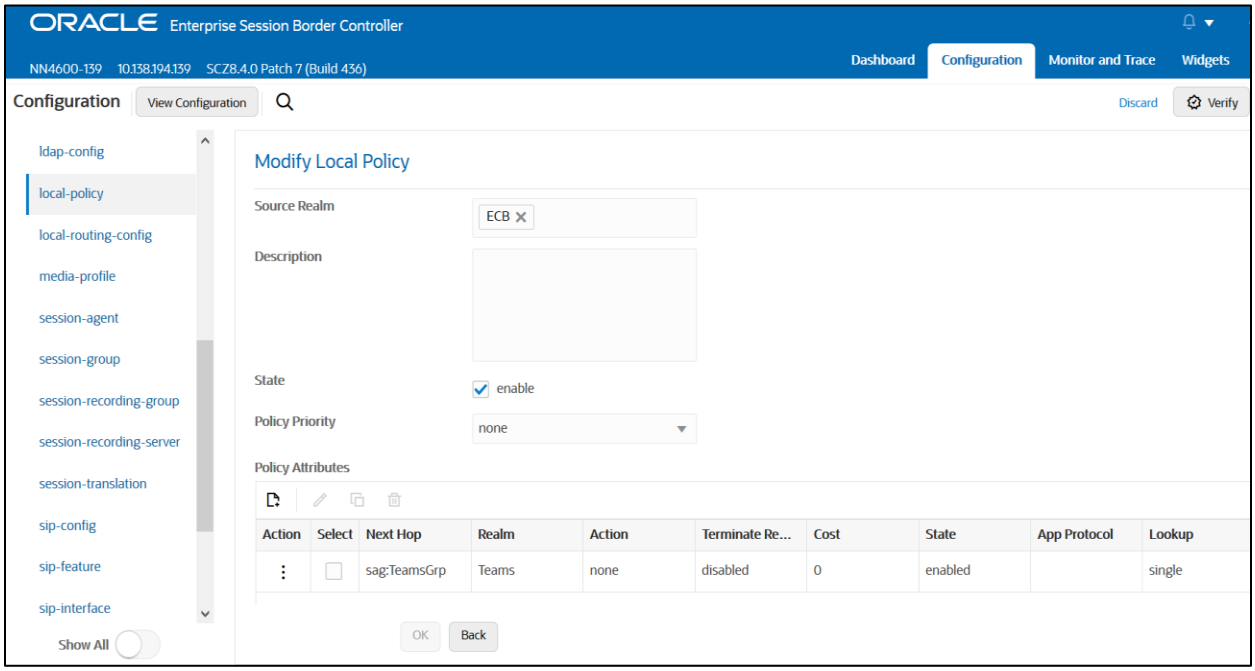
To route the calls from Teams to ECB Realm, use the below local –policy
Please note that the next hop is ECB IP which is 10.232.50.70





To route the calls from ECB to Teams Realm, use the below local –policy

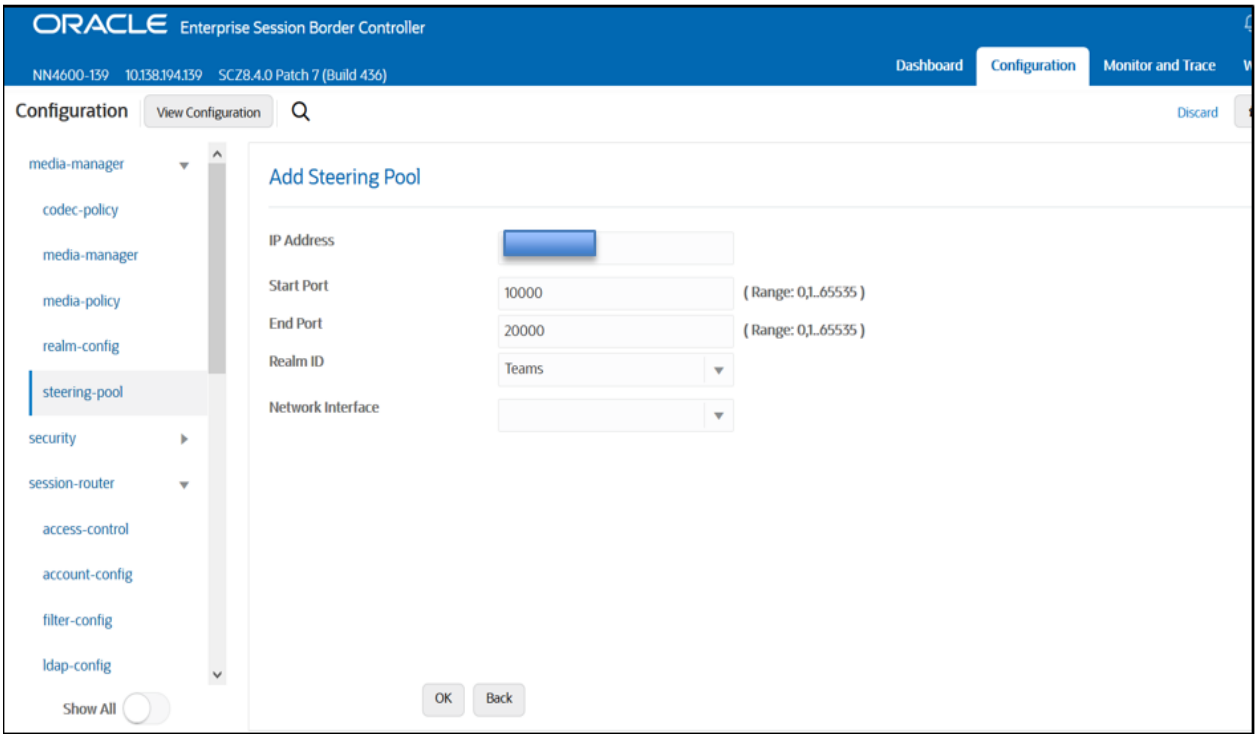




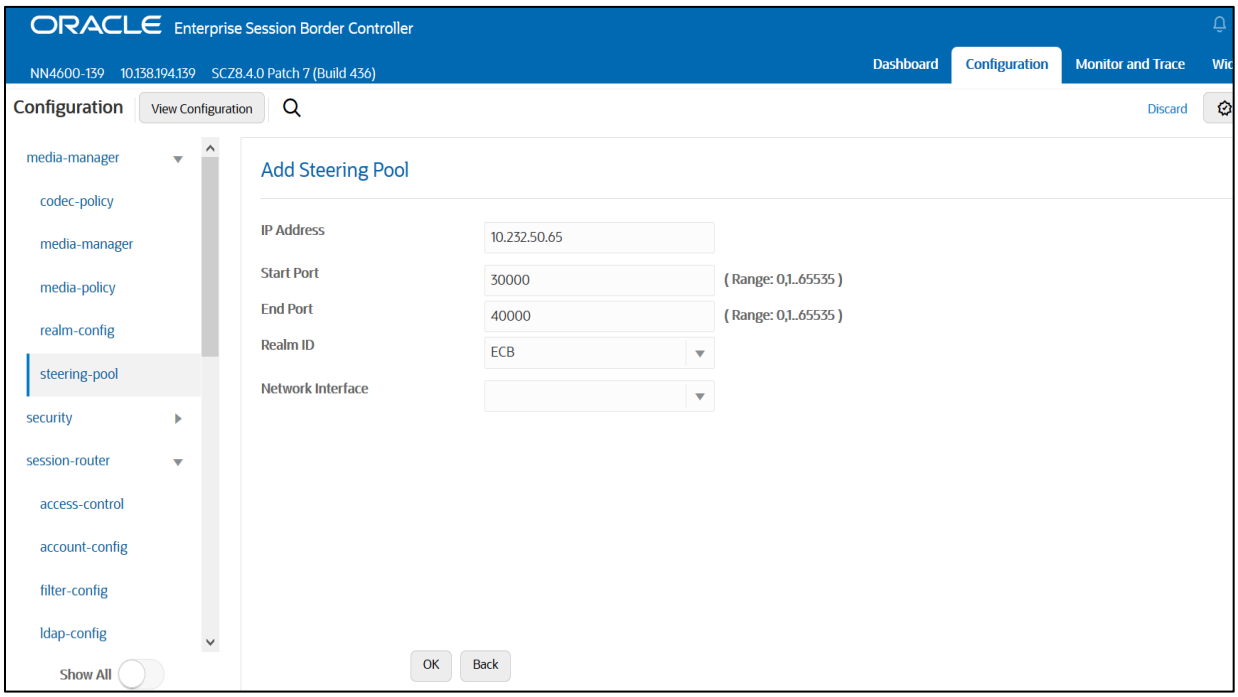
6.12. Configure steering-pool

Steering-pool config allows configuration to assign IP address(es), ports & a realm.

Teams side steering pool.



ECB side steering pool.



With this, SBC configuration is complete

7. Existing SBC configuration

If the SBC being used is an existing SBC with functional configuration, following configuration elements are required:

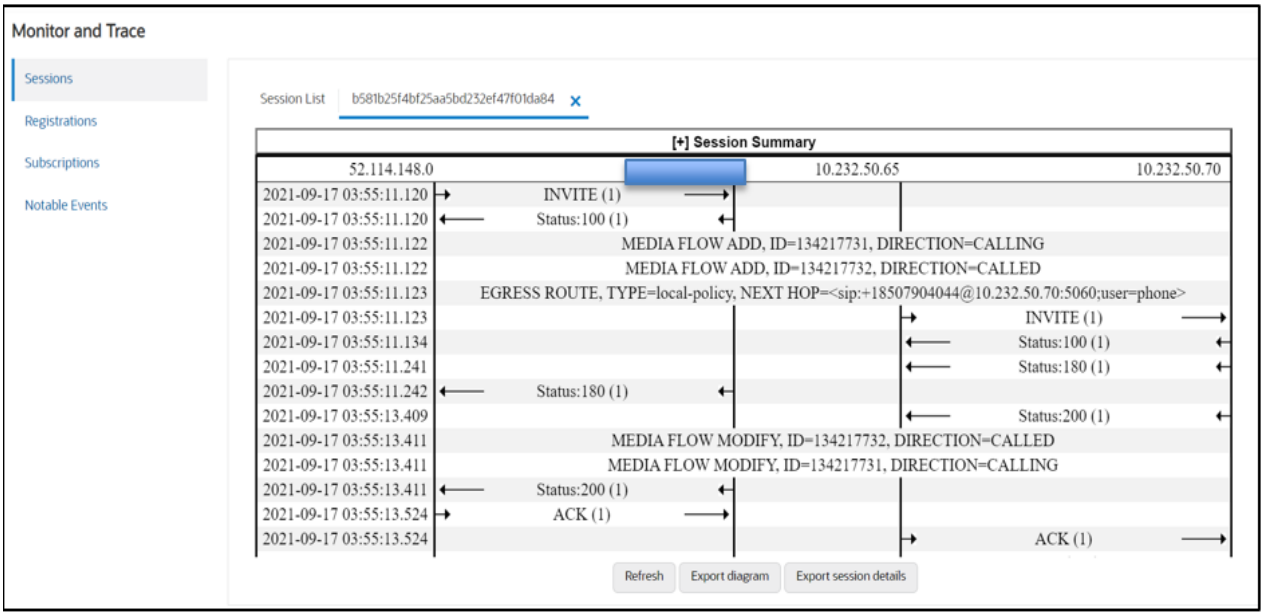
- New realm-config
- New sip-interface
- New session-agent
- New steering-pools
- New local-policy

Please follow the steps mentioned in the above chapters to configure these elements.

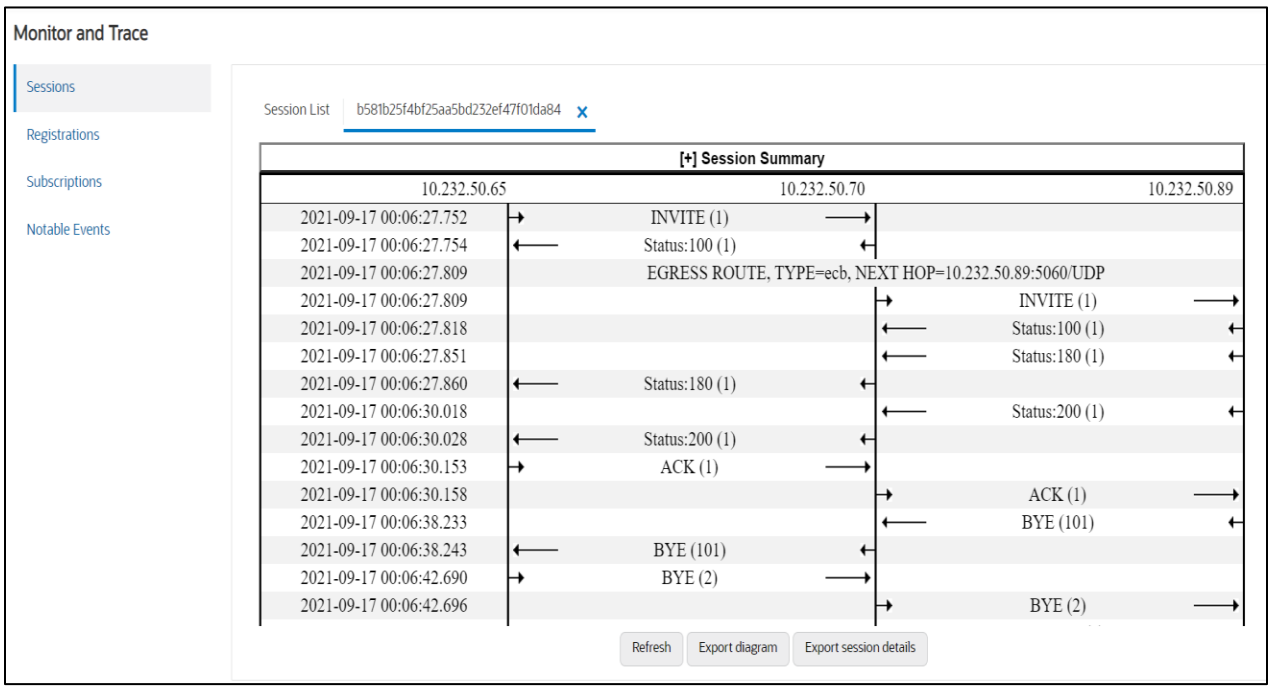
8. Verification of Sample Call flows

Once the configuration is complete, we can try making sample calls and can check the signaling path and the call trace details as below:

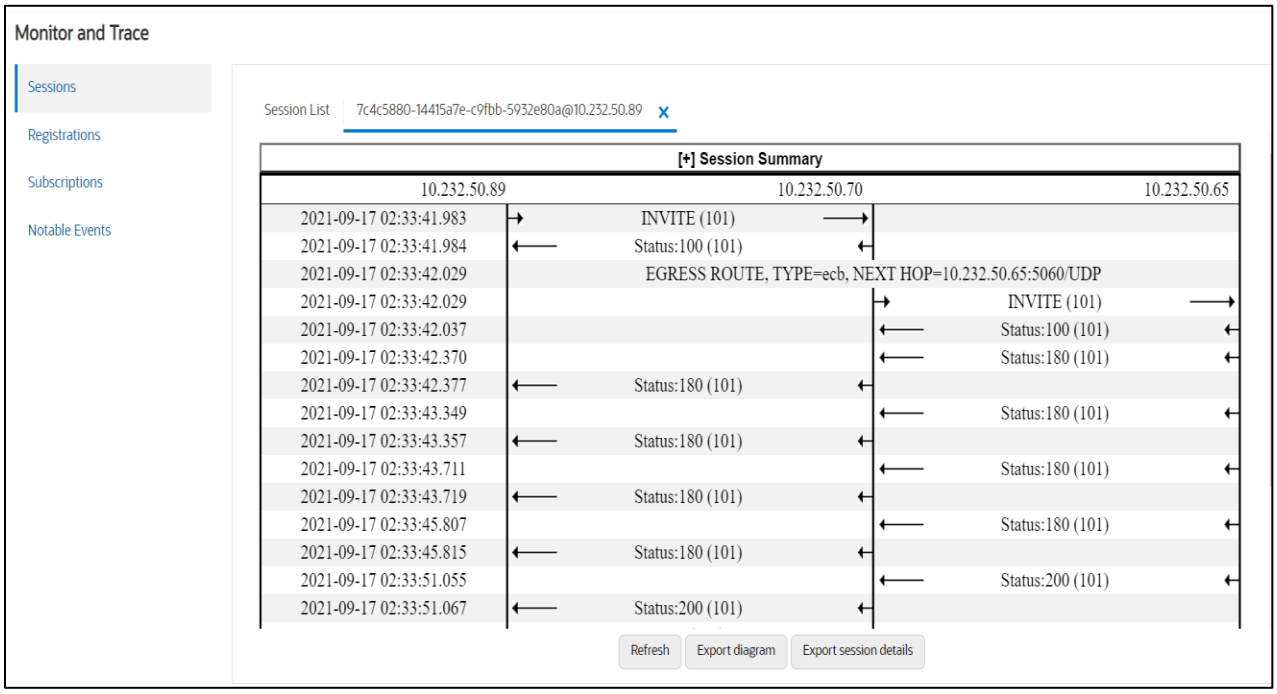
1. Make Call from Teams User to IP-PBX user (CUCM user) using ECB LDAP config and check the call flow. The Call from Teams reaches SBC and then routed to ECB IP in below trace.



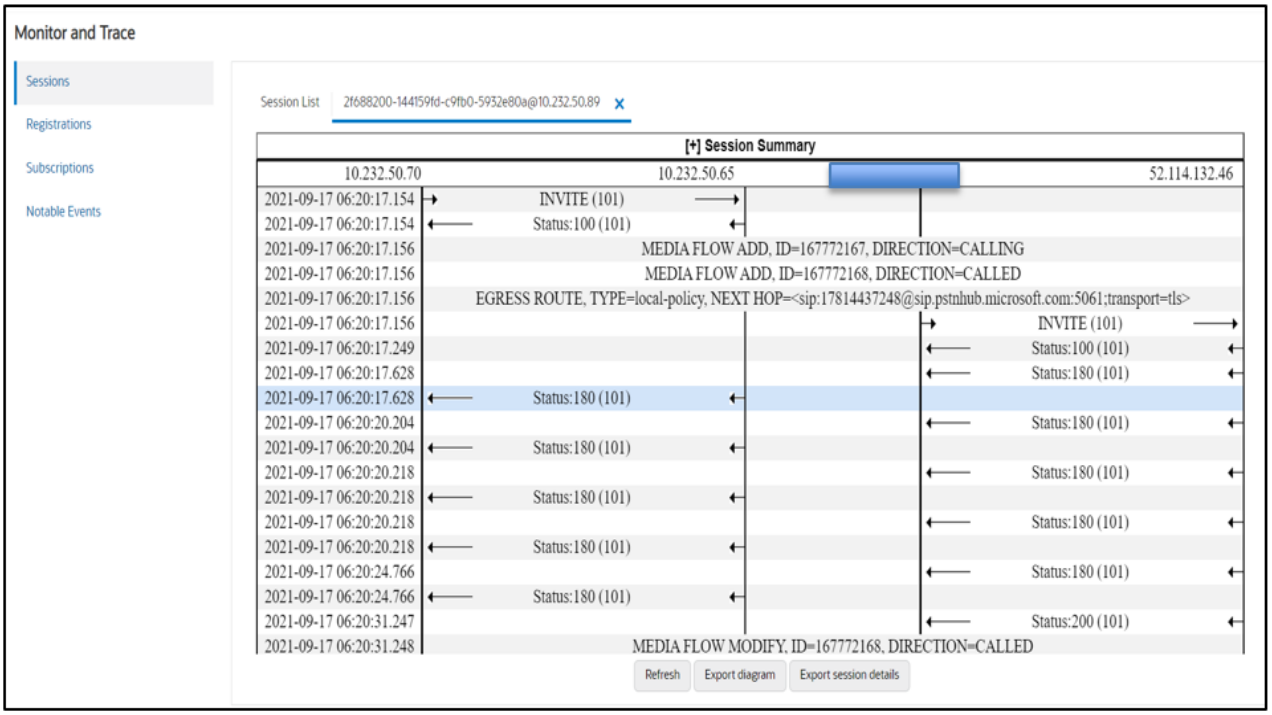
The call reaches ECB IP and then ECB does LDAP search and then routes the call to CUCM IP. Then the call reaches CUCM and then the call is established as in below trace.



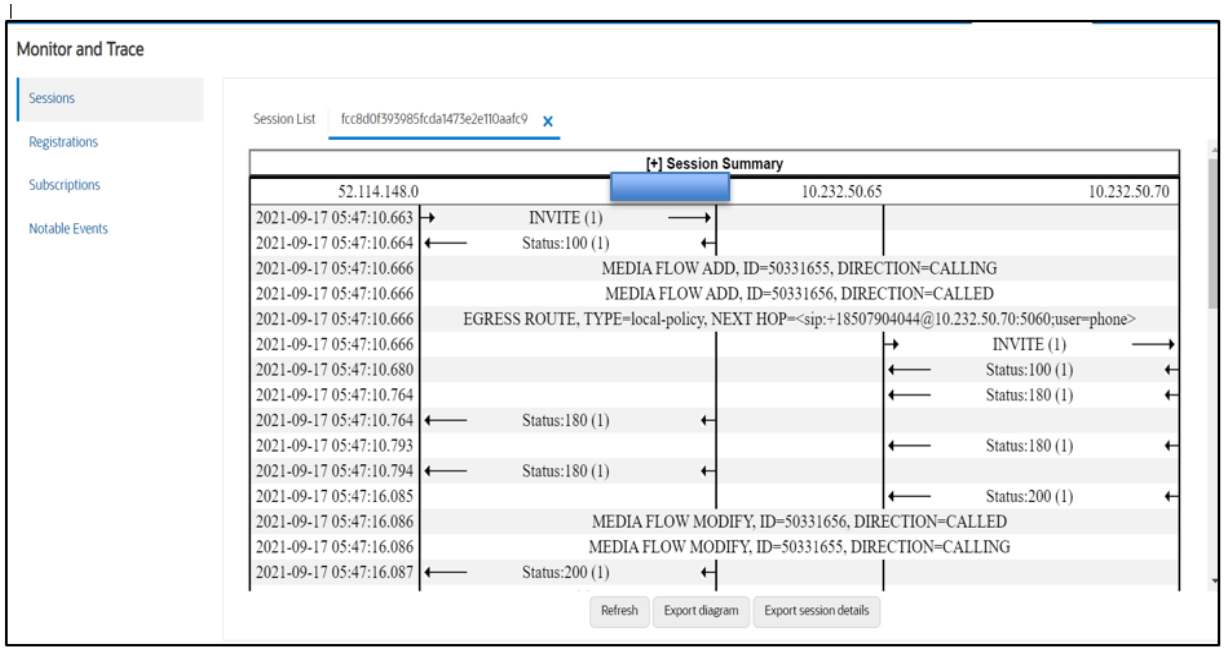
2. Make Call from CUCM user to Teams User using ECB LDAP config and check the call flow. The Call from CUCM user reaches ECB first and ECB does LDAP query and sends call to SBC interface IP as given below.



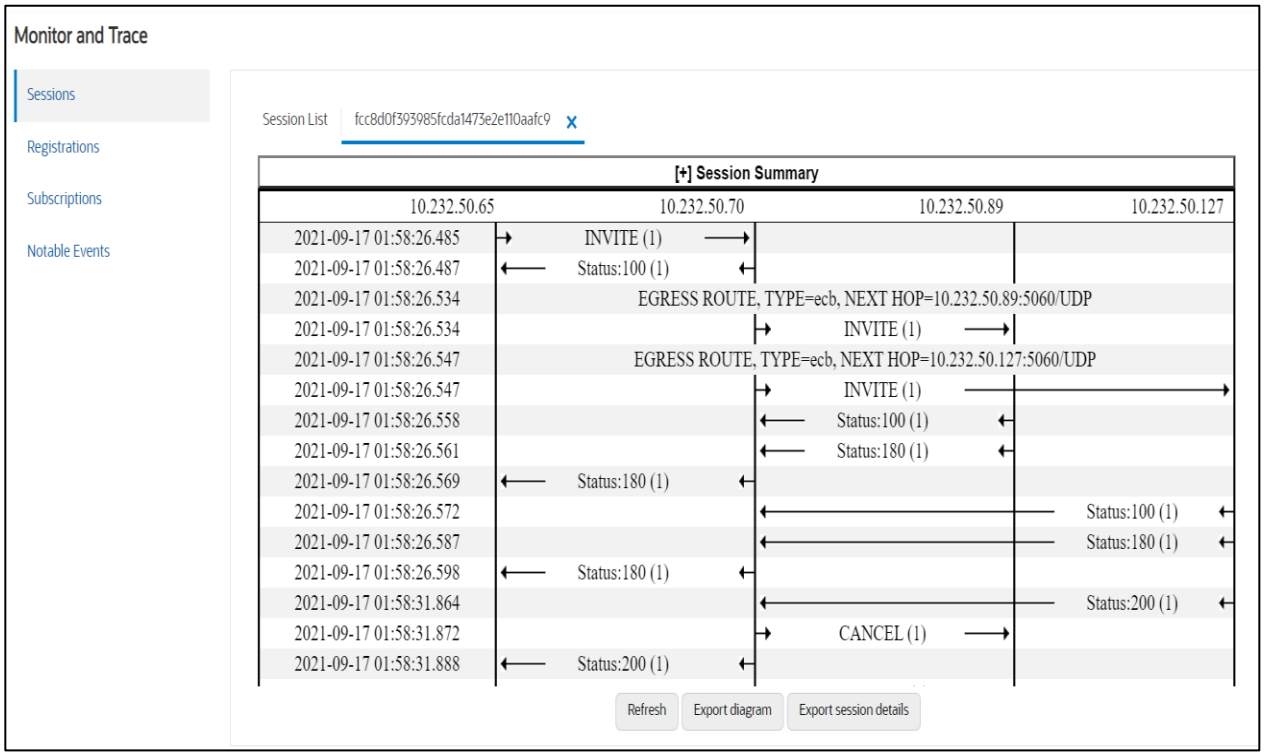
When Call reaches SBC IP, Call is then routed to Teams User and the call gets established as shown in the below trace.

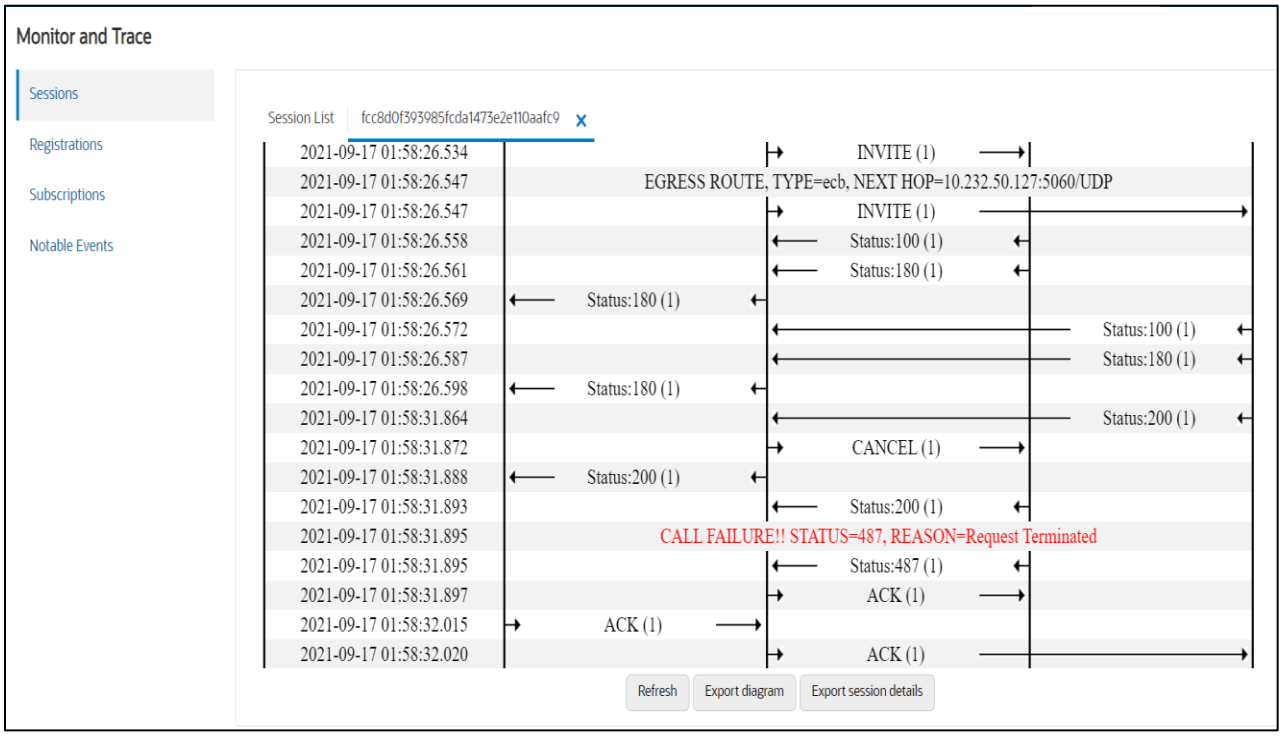


3. Make Call from Teams User to On premise IP-PBX user and check the ECB LDAP parallel forking call flow as shown below. The Call from Teams User reaches SBC and then routed to ECB as shown below.



The Call now reaches ECB and it does LDAP parallel forking and sends calls to both Avaya and CUCM User and when Avaya User attends the call, ECB sends CANCEL to CUCM user as shown below.





We can also make calls using ECB ATG config parallel forking using the configuration given in this document. The call trace looks similar to the above call trace from SBC and ECB point of view and so we did not add the call trace for the ECB ATG config scenario.

We can also make calls to auto-attendant (Either to Teams – towards SBC side or to On Premise IP-PBX - Core side) as one of the DIDs along with other normal Phone number DIDs using the ECB ATG parallel forking configuration. We need to add auto-attendant DID in the User Entry of the ECB config. Only thing to remember here is we need to assign the same number assigned to auto-attendant to other PBX users involved in ECB ATG parallel forking configuration.

Appendix A

Following are the test cases that are executed as part of Oracle ECB and SBC integration interworking with on premise IP-PBX and MS Teams

Serial Number	Test Cases Executed	Result
1	Teams User calling On premise IP-PBX user (Avaya/CUCM/Genesys) using ECB LDAP feature	Pass
2	On premise IP-PBX user (Avaya/CUCM/Genesys) calling Teams User using ECB LDAP feature	Pass
3	Teams User calling On premise IP-PBX user (Avaya/CUCM/Genesys) using ECB LDAP parallel forking feature	Pass
4	Testing Oracle ECB ATG parallel forking by making calls to Core side User (On Premise IP-PBX User) and SBC side user (Teams User)	Pass
5	Testing Oracle ECB ATG parallel forking by making calls to Core side User (On Premise IP-PBX User) and Teams side auto attendant	Pass
6	Testing Oracle ECB ATG parallel forking by making calls to Core side User and voice mail profile (On Premise IP-PBX User)	Pass

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

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

ORACLE

CONNECT WITH US

blogs.oracle.com/oracle
facebook.com/Oracle/
twitter.com/Oracle
oracle.com

Integrated Cloud Applications & Platform Services