Business Services

TECHNICAL GUIDE to access Business Talk & BTIP Cisco CUCM

versions addressed in this guide: 12.0 & 12.5

Version of 07/10/2019

Table of contents

1 Goal of this document **5**
2 Architecture overview **6**
2.1 CUCM without CUBE 6
2.2 CUCM with CUBE (Cisco Unified Border Element) 7
2.3 CUCM with Oracle SBC (Session Border Controller) 8
2.3.1 Unsecured SIP Trunk 9
2.3.2 Secured SIP Trunk 10
3 Parameters to be provided by customer to access service **11**
3.1 CUCM without CUBE 11
3.2 CUCM with CUBE (flow through) 11
3.3 CUCM with Oracle SBC 12
4 Certified software and hardware versions **13**
4.1 CUCM certified versions 13
4.2 CUCM certified applications and devices versions 13
4.3 CUBE certified versions 14
4.4 Oracle ESBC certified versions 14
5 Cisco Call Manager configuration **15**
6 Cisco Unity Connection configuration **30**
7 Unified Contact Center Express configuration **31**
7.1 Provisioning UCCX (CUCM part) 31
7.1.1 Adding agents 31
7.1.2 Activation and Configuring IP Phone Agent service 32
7.1.3 UCCX Application Users on CUCM 32
7.2 UCCX part of configuration 33
7.2.1 Provisioning Call Control Group (CCC) 33
7.2.2 Resources and assignment of skills 33
7.2.3 Configuring Customer Service Queues (CSQ) 33
7.2.4 Application and Script configuration 34
7.2.5 Trigger configuration 34
8 Cisco Unified Attendant Console configuration **35**
9 CUCM with Cisco Unified Border Element configuration **38**
9.1 General CUBE configuration (flow-through mode by default) 38
9.2 Configuration for a CUCM cluster and two CUBEs 39
9.3 Configuration for a single CUCM server and one CUBE 42
9.4 Configuration for a CUCM cluster and one CUBE 44
9.5 Design for Local SIP Trunking 46
9.5.1 Region configuration 46
9.5.2 Device Pool configuration 47
9.5.3 Route List configuration 48
9.5.4 Route Group Configuration 48
9.5.5 Locations (Call Admission Control) 48
9.5.6 SIP Trunk Configuration 49
10 CUCM with Oracle Session Border Controller configuration **50**
10.1 CUCM configuration 50
10.2 Oracle SBC configuration 55
10.2.1 Oracle SBC information required for CUCM interconnection 55

10.2.2 Oracle SBC information required for a new IPBX ...55
10.2.3 Information required for BTIP / Btalk SIP Infrastructure ...56
10.2.4 SBC Object naming convention ...56
10.2.5 Certificate ...57
10.2.6 Licenses & ESBC entitlement setup ...57

11 Expressway ... 58

11.1 Architecture overview ...58
11.2 Call Flows ...58
11.3 Endpoint Authentication & Encryption ...59
11.3.1 Authentication ...59
11.3.2 Directory integration ...59
11.3.3 Telephony features ...60
11.4 CUCM configuration update ...61
11.5 Expressway specific configuration ...61

12 Fax ... 65

12.1 Configuration for BT/BTIP SIP trunking ...65
12.1.1 T.38 global settings ...65
12.1.2 Codec configuration ...65
12.1.3 Example of VoIP dial-peer configuration ...65
12.1.4 POTS dial-peer ...66
12.1.5 CUCM Configuration ...66
12.1.6 CUBE Configuration ...68
12.1.6.1 Media Passing through CUBE (media flow-through vs. media flow-around) ...69
12.1.6.2 Codecs ...69
12.1.6.3 SIP user agent ...69
12.2 Integrating Sagem XMedius Fax Server Enterprise 8.0 with CUCM ...70
12.2.1 Highlights for Sagem XMediusFax Server Enterprise 8.0.0.300: ...70
12.2.2 Supported fax features with BTIP Service ...71
12.3 Sagem XMediusFax Server components configuration ...71
12.3.1 CUCM Configuration ...80
12.3.1.1 SIP Trunk Configuration ...80
12.3.1.2 Route Pattern Configuration ...81
Confirmation tests ...82
12.4 Validation overview ...82
12.5 Validation ...83
12.5.1 Functional ...83
12.5.2 Statistical ...83

ANNEX A: Provisioning Oracle ESBC ... 84

1.1 Global configuration ...84
1.1.1 Media configuration ...84
1.1.1.1 Media Manager Configuration ...84
1.1.2 Codec Policy ...85
1.1.2.1 Media Security Policy ...86
1.1.3 Global Sip Configuration ...88
1.1.3.1 User-Agent ...88
1.1.3.2 Sip-config ...88
1.1.3.3 Header Whitelists ...90
1.1.3.4 SIP enforcement Profile ...93
1.1.3.5 SIP features ...94
1.1.3.6 Response maps ...96
1.2 Business Talk/ BTIP OBS Carrier North SIP configuration for Oracle ESBC configuration ...98
1.2.1 Unsecured SIP Trunk through UDP ...98
1.2.1.1 Core realm Configuration ...98
1.2.1.2 Core realm sip-interface ...101
1.2.1.3 Steering-pool Configuration ...103

1.2.2 Secured SIP Trunk through TLS 104
1.2.2.1 SBC Certfiicate 104
1.2.2.2 Customer CA certificate(s) 106
1.2.2.3 TLS profile 108
1.2.2.4 SRTP configuration 108
1.2.2.4.1 SDES profile 108
1.2.2.4.2 Media-sec-policy 109
1.2.2.5 Core realm Configuration 110
1.2.2.6 Core realm sip-interface 113
1.2.2.1 Steering-pool Configuration 116
1.2.3 BT/BTIP objects 117
1.2.3.1 Nominal Session agent 117
1.2.3.2 Backup Session Agent 119
1.2.3.3 Session Agent Groups 122
1.2.3.3.1 Nominal Session Agent Group 122
1.2.3.4 Access List 123
1.2.3.5 BT Nominal Session Agent- control 123
1.2.3.6 BT Backup Session Agent- control 124
1.2.4 Provisioning BT/BTIP on a backup ESBC 124
1.2.5 Local-policy from core to access 125
1.3 Customer Cisco CUCM IPBX South SIP configuration for Oracle SBC configuration 128
1.3.1 Provisioning a Cisco CUCM IPBX on the ESBC 128
1.3.1.1 Access Network interface 128
1.3.1.2 Access Realm 129
1.3.1.3 Access Steering-pool 132
1.3.1.4 Access sip-interface 132
1.3.2 Provisioning a new customer Cisco IPBX on a backup ESBC 135
1.3.3 Cisco IPBX objects 135
1.3.3.1 Nominal Session agent 135
1.3.3.2 Backup Session Agent 137
1.3.3.3 Session Agent Groups 140
1.3.3.3.1 Nominal Session Agent Group 140
1.3.3.3.2 Backup Session Agent Group 140
1.3.3.4 Access List 142
1.3.3.5 PBX Nominal Session Agent- control 142
1.3.3.6 PBX Backup Session Agent- control 143
1.3.4 Local-policy from access to core 144
1.4 SIP manipulations 145
1.4.1 outToPBXsipManip 146
1.4.2 outToBT 148

1 Goal of this document

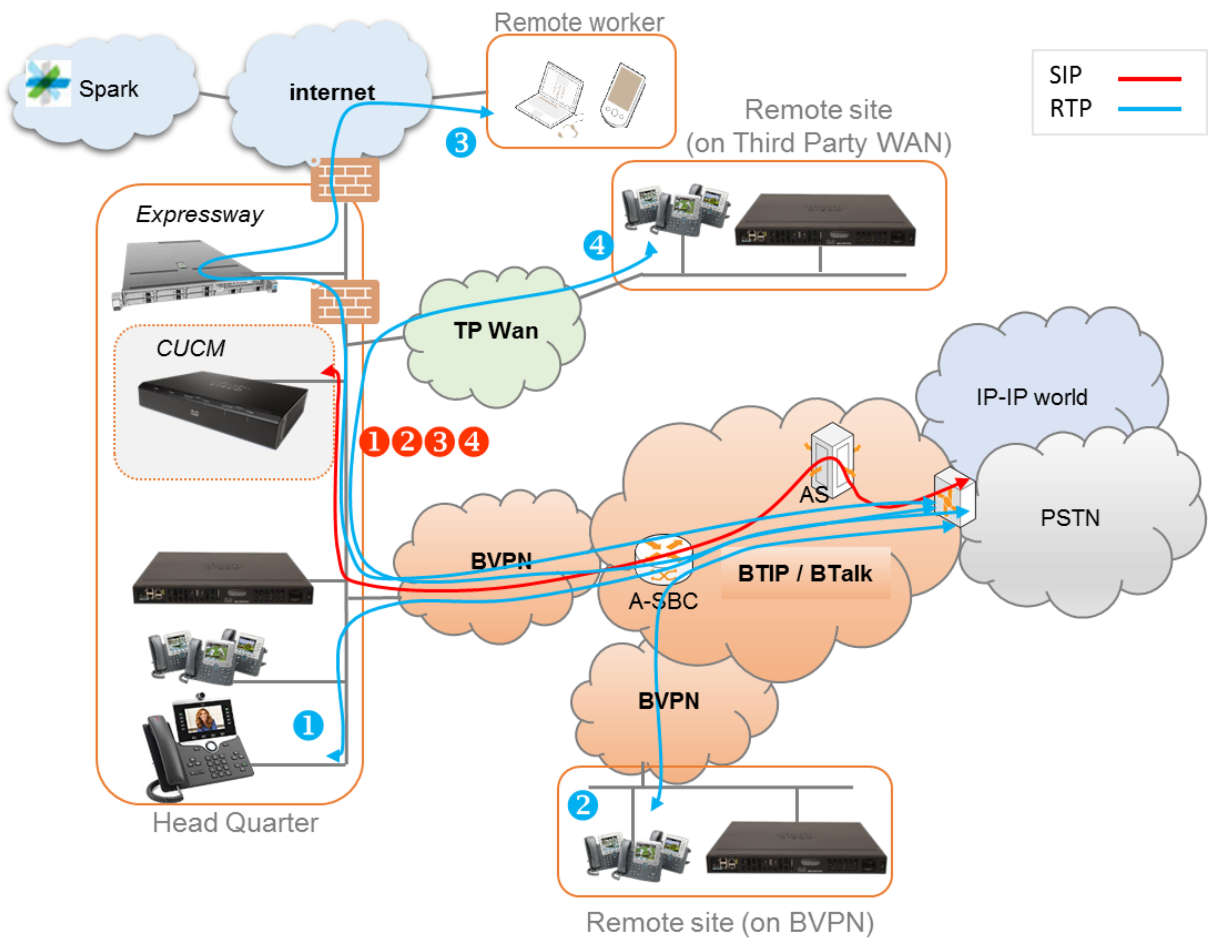
The aim of this document is to list technical requirements to ensure the interoperability between Cisco CUCM IPBX with Business Talk IP SIP, hereafter so-called “service”.

Note:

- This document describes “only” the main supported architectures either strictly used by our customers or that are used as reference to add specific usages often required in enterprise context (specific redundancy, specific ecosystems, multi-PBX environment, multi-codec and/or transcoding, recording…)

2 Architecture overview

2.1 CUCM without CUBE



Notes :

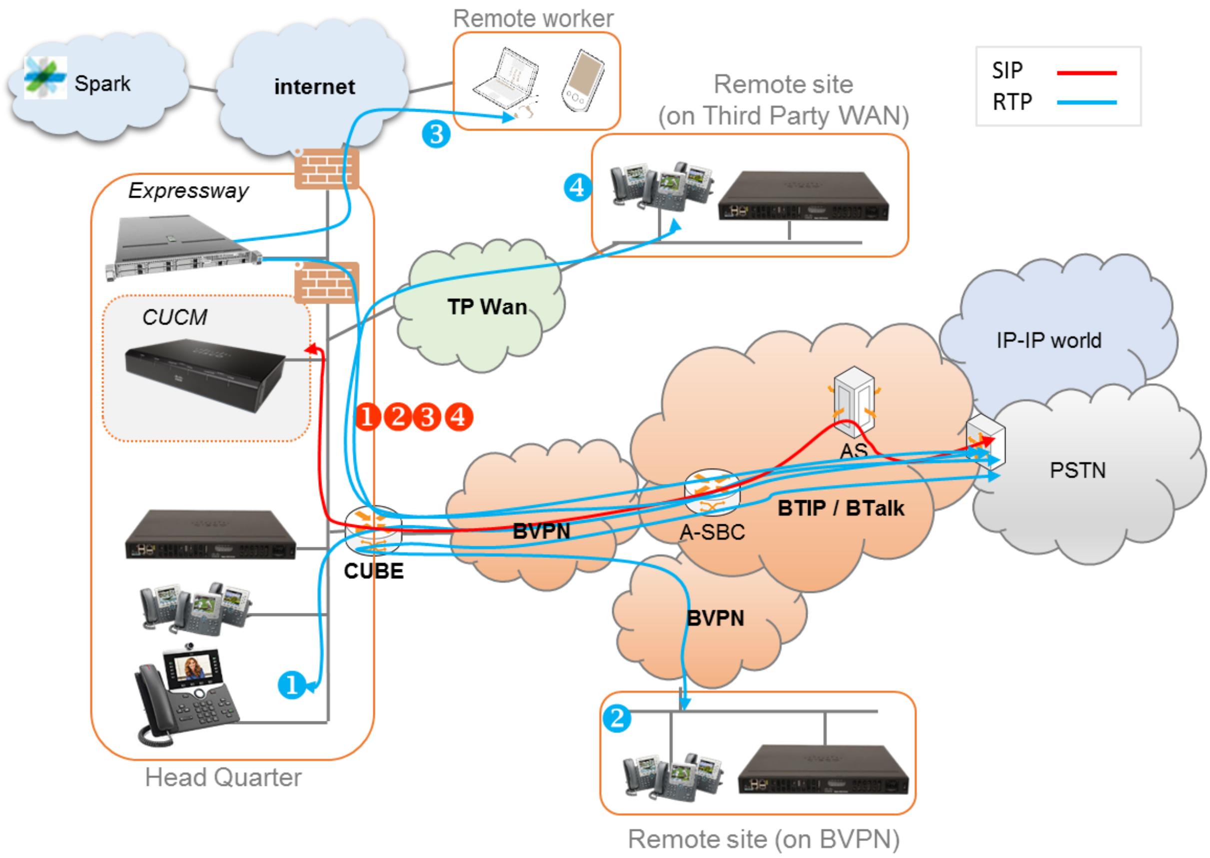
- in the diagram above, the SIP, proprietary and Spark internal flows are hidden.
- call flows will be the similar with or without CUCM redundancy

In this architecture :

- all 'SIP trunking' signaling flows are carried by the CUCM server and routed on the main BVPN connection.
- Media flows are direct between endpoints and the Business Talk/BTIP but IP routing differs from one site to another :
  - For the Head Quarter site, media flows are just routed on the main BVPN connection
  - For Remote sites on BVPN, media flows are just routed on the local BVPN connection (= distributed architecture),

- For Remote sites on Third Party WAN, media flows are routed through the Head Quarter (but not through the IPBX) and use the main BVPN connection (= **centralized architecture**).

2.2 CUCM with CUBE (Cisco Unified Border Element)

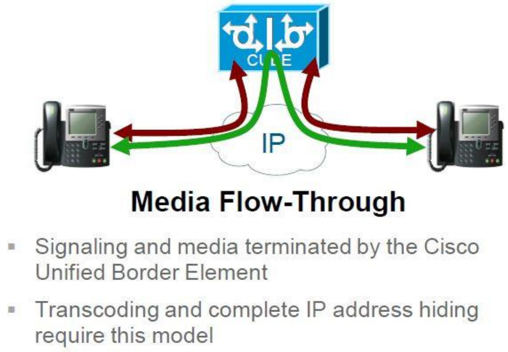


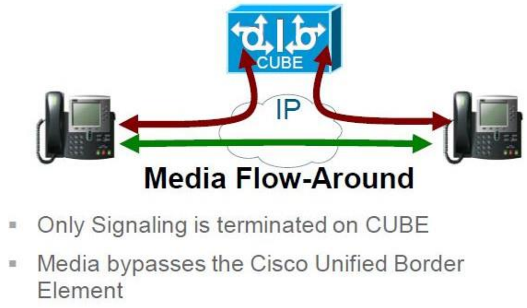
Notes :

- in the diagram above, the SIP, proprietary and Spark internal flows are hidden.
- call flows will be similar with or without CUCM redundancy.

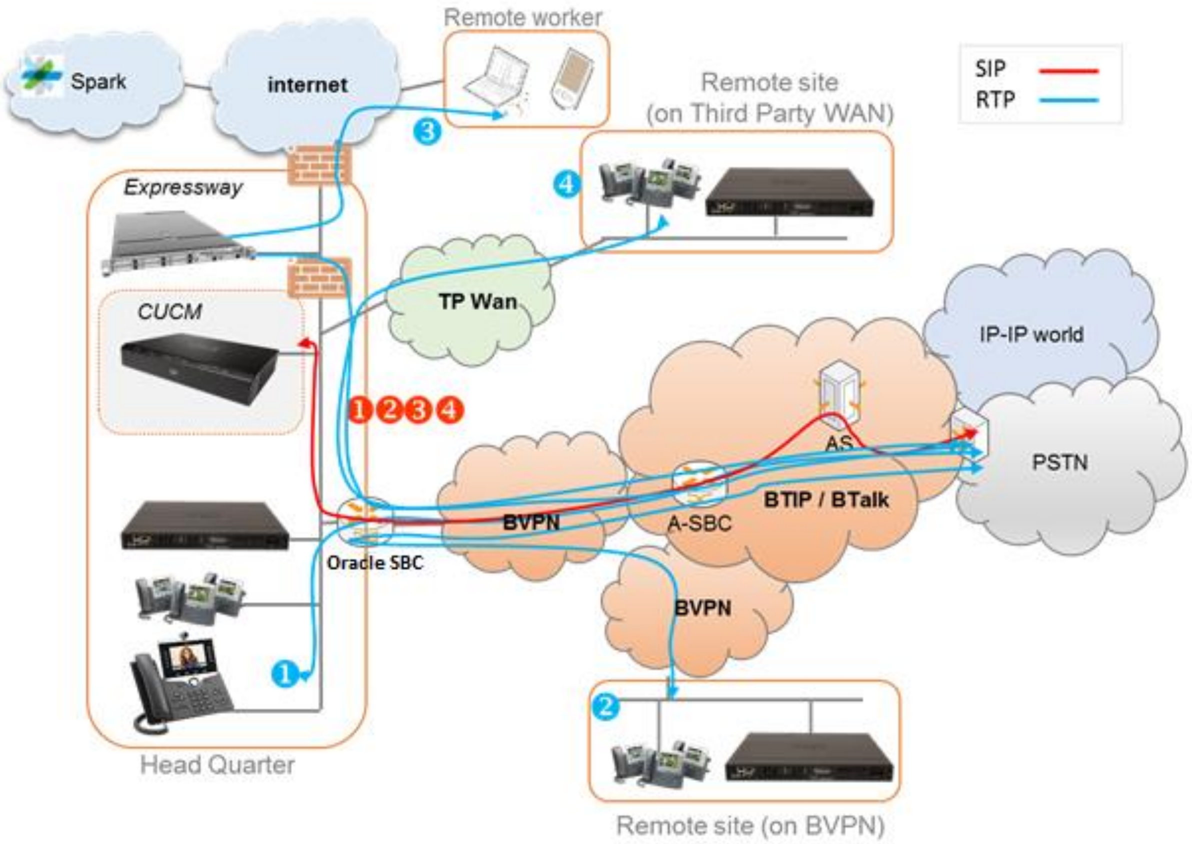
In this architecture, all SIP trunks are anchored by the CUBE but with 2 modes for the media :

- "Flow-through" mode → signalling and media flows cross the CUBE.
- "Flow-around" mode → signaling flows cross the CUBE, but media flows go directly towards endpoints





2.3 CUCM with Oracle SBC (Session Border Controller)



In this architecture, all SIP trunks are anchored by the Oracle Enterprise SBC. The call flows are very similar to the architecture with Cisco CUBE. Session Border Controller is mostly transparent for SIP traffic. It can also be used for TLS encryption ensuring secure traffic between Oracle ESBC and Orange SBC.

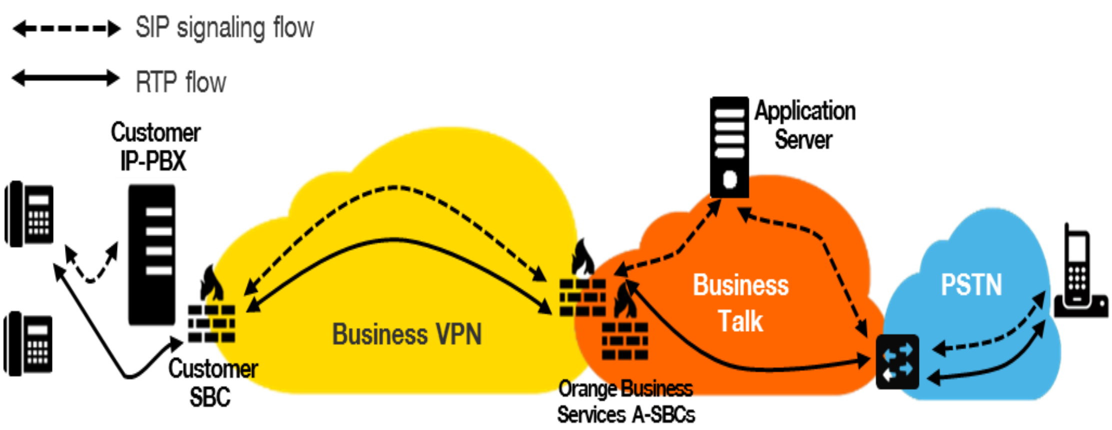
Oracle Enterprise SBC v.8.2 has been validated with Cisco CUCM v.12.0.

The following features have been tested for CUCM with Oracle SBC integration:

- Basic Telephony features (basic calls, CLIR, forward, transfer, MoH, DTMF)
    - IP Phones

  - FXS Gateway for analog phones
- Fax
  - Sagem Xmedius Fax server
  - SIP Fax on FXS Gateway
- TLS Encryption between Oracle ESBC and Orange SBC

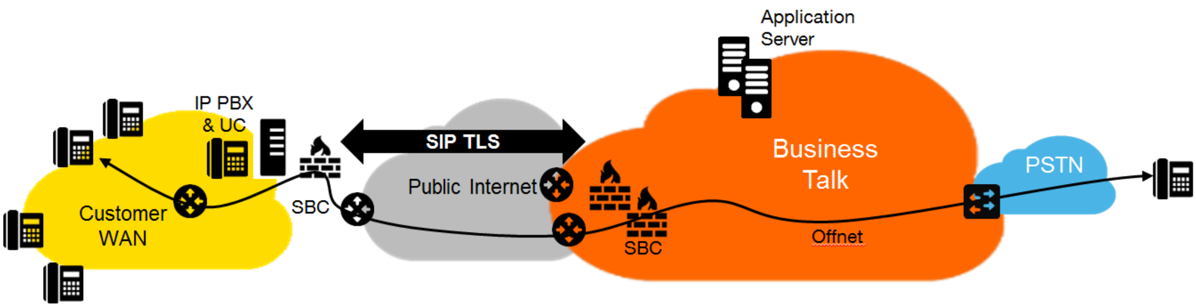
2.3.1 Unsecured SIP Trunk



In this architecture :

- Both 'SIP trunking' and RTP media flows between endpoints and the Business Talk/BTIP are anchored by the "customer SBC". For the Head Quarter & remote sites sites, media flows are routed through the SBC and the main BVPN connection.
- Both 'SIP trunking' on North (OBS Carrier) and South side of the SBC must be configured in "clear" mode though UDP.

2.3.2 Secured SIP Trunk



In this architecture :

- both 'SIP trunking' and RTP media flows between endpoints and the Business Talk/BTIP are anchored by the "customer SBC". For the Head Quarter & remote sites sites, media flows are routed through the SBC then Internet.
- 'SIP trunking' on North (OBS Carrier) side of the SBC must be configured in "secured" mode though TLS encryption and media.

3 Parameters to be provided by customer to access service

IP addresses marked in red have to be indicated by the customer, depending on customer architecture scenario.

3.1 CUCM without CUBE

Head Quarter (HQ) or Branch Office (BO) architecture	Level of Service	Customer IP addresses used by service	
		Nominal	**Backup**
CUCM Business Edition (1 server)	No reduncdancy (1 Publisher)	CUCMBE IP@	N/A
CUCM (1 Publisher + 1 Subscriber)	Local redundancy Subscriber (Nominal) / Publisher (Backup) Publisher and Subscriber are on different servers)	Subscriber IP@	Publisher IP@
CUCM (1 Publisher + 2 Subscribers) Subscribers Nominal/Backup	- Local redundancy Subscriber1 (Nominal) / Subscriber2 (Backup) - If more than 1 Subscriber, the SIP trunks are held by the Subscribers. The Publisher holds the database.	Subscriber1 IP@	Subscriber2 IP@
CUCM (1 Publisher + 2 Subscribers) Subscribers Load Sharing	- Local redundancy and Load Sharing Subscriber1 / Subscriber2 - The Subscribers share the load in a round robin fashion (Also applicable with N Subscribers)	Subscriber1 IP@ Subscriber2 IP@	N/A
CUCM with clustering over WAN (1 Publisher + 1 Subscriber)	- Site redundancy: Subscriber and Publisher servers hosted by 2 different physical sites	Subscriber IP@	Publisher IP@
CUCM with clustering over WAN (1 Publisher + 2 Subscribers) Subscribers Nominal/Backup	- Site redundancy: the 2 Subscribers are hosted by 2 different physical sites (Subscriber1(Nominal) / Subscriber2(Backup)) - If more than 1 Subscriber, the SIP trunks are held by the Subscribers. The Publisher holds the database.	Subscriber1 IP@	Subscriber2 IP@
CUCM with clustering over WAN (1 Publisher + 2 Subscribers) Subscribers Load Sharing	- Site redundancy: the 2 Subscribers are hosted by 2 different physical sites (Subscriber1 + Subscriber2) - The Subscribers share the load in a round robin fashion	Subscriber1 IP@ Subscriber2 IP@	N/A
		Nominal	**Backup**
Remote site without survivability	No survivability, no trunk redundancy	N/A	N/A
SRST	Local site survivability and trunk redundancy via PSTN only	N/A	N/A

3.2 CUCM with CUBE (flow through)

Head Quarter (HQ) or Branch Office (BO) architecture	Level of Service	Customer IP addresses used by service	
		Nominal	**Backup**
CUCM + Single CUBE	No redundancy	CUBE IP@	N/A
CUCM + 2 CUBES **warning:** - Site access capacity to be sized adequately on the site carrying the 2nd CUBE in case both CUBEs are based on different sites	**- Local redundancy:** if both CUBES are hosted by the same site (CUBE1+CUBE2) **- Geographical redundancy:** if each CUBE is hosted by different sites (CUBE1+CUBE2)	CUBE1 IP@	CUBE2 IP@
		Nominal	**Backup**
Remote site without survivability	No survivability, no trunk redundancy	N/A	N/A
SRST	Local site survivability and trunk redundancy via PSTN only	N/A	N/A

3.3 CUCM with Oracle SBC

Head Quarter (HQ) or Branch Office (BO) architecture	Level of Service	Customer IP addresses used by service	
		Nominal	Backup
CUCM + Oracle SBC	No redundancy	Oracle IP@	N/A
CUCM + 2 Oracle SBC Nominal / Backup mode	- **Local redundancy**: both SBC are hosted on the same site OR - **Geographical redundancy** both SBC are hosted on 2 different sites	Oracle IP@	Oracle2 IP@
CUCM + 2 Oracle SBC Load Sharing	- **Local redundancy**: both SBC are hosted on the same site OR - **Geographical redundancy** both SBC are hosted on 2 different sites	Oracle IP@	Oracle2 IP@
CUCM + 2 Customer SBC HA mode	**- Local redundancy:** both SBC are hosted on the same site OR **- Geographical redundancy** both SBC are hosted on 2 different sites **warning:** Link level 2 between SBC with max delay 50ms required for geo-redundancy	Oracle Virtual IP@	N/A

4 Certified software and hardware versions

4.1 CUCM certified versions

Cisco IPBX			
Equipment	Equipment Version	validation status	IPBX Version
CUCM CBE5000/6000	R12.0	✓	Load 12.0.1.21900-7 min
	R12.5	✓	Load 12.5.1.10000-22 min

4.2 CUCM certified applications and devices versions

Cisco ecosystems					
Equipment		Equipment Version	validation status	IPBX Version	Comment
Attendant Console	CUxAC	12.0.x	✓	R12.0	Standard and Advanced editions
				R12.5	
Voice Mail	Unity Connection	12.0.1000-6	✓	R12.0	
		12.5	✓	R12.5	
	Unity Express	12.0.x	✓	R12.0	
Contact center	UCCX	12.0.x	✓	R12.0	
MGW	Cisco IOS Cascaded MediaGateway (ISR 28xx/38xx)		not supported	R12.0	
			not supported	R12.5	
	Cisco IOS Cascaded MediaGateway (ISR 29xx/39xx)	15.7(3)M	✓	R12.x	
	Cisco IOS Cascaded MediaGateway (ISR 43xx/44xx)	16.6.3	✓	R12.0	
		16.9.4	✓	R12.5	
	Analog GW Cisco ATA187		not supported	R12.x	
	Audiocodes MP112 FXS		on demand	R12.x	
	Analog GW Cisco VG 224		not supported	R12.x	
	Analog GW Cisco VG 202-204		not supported	R12.x	
	Analog GW Cisco VG 202-204 XM	15.5(3)M2	✓	R12.x	
	Analog GW Cisco VG 310-320-350	15.7(3)M	✓	R12.x	
	Analog GW Cisco ATA190	1.2.1(004)	✓	R12.0	
		1.2.2(003)	✓	R12.5	
VOIP	Cisco VoIP GW		on demand	R12.x	
	OneAccess VoIP GW (Business Livebox)		on demand	R12.x	

Phones	Cisco Unified Communication Manager Assistant (IPMA)		not supported	R12.x	
	All Cisco SCCP phones (skinny)		✓	R12.x	
	All Cisco SIP phones		✓	R12.x	
	IPCommunicator SCCP		not supported	R12.x	
	Jabber	11.9.3	✓	R12.x	
	CUCILync		✓	R12.x	
	IP DECT ASCOM		✓	R12.x	

Third Party Equipments	Conecteo KIAMO	6.1	✓	R11.x R12.0	Dorsal mode

4.3 CUBE certified versions

Cisco CUBE				
Equipment	Equipment Version	validation status	IPBX Version	Comment
Cisco Unified Border Element (CUBE) - “flow thru” mode	16.6.3	✓	R12.0	
	16.9.4	✓	R12.5	
Cisco Unified Border Element (CUBE) - “flow around” mode	16.6.3	✓	R12.0	
	16.9.4	✓	R12.5	

4.4 Oracle ESBC certified versions

Oracle ESBC				
Equipment	Equipment Version	validation status	IPBX Version	Comment
Oracle Enterprise Session Border Controller	8.2 Patch 2 (Build 58)	✓	R12.0	

5 Cisco Call Manager configuration

The checklists below present all the configuration steps required for interoperability between the service and CUCM.

Cisco Call Manager Service **Codec and payload configuration**	
Menu	**Value**
System > Service Parameters > Appropriate server > Cisco CallManager (Active) > Advanced > Clusterwide Parameters (System – Location and Region)	
Preferred G.711 Millisecond Packet Size	20
Preferred G.729 Millisecond Packet Size	20
G.722 Codec Enabled	Enabled for All Devices
Cisco CallManager Service Codec and payload configuration	
System > Service Parameters > Appropriate server > Cisco CallManager (Active) > Advanced Clusterwide Parameters (Service)	
Duplex Streaming Enabled	True
Media Exchange Timer	5
Silence suppression	False
Silence suppression for Gateways	False
Media Exchange Timer	True
Cisco CallManager Service SIP Parameters	
System > Service Parameters > Appropriate server > Cisco CallManager (Active) > Advanced Clusterwide Parameters (Device - SIP)	
Retry Count for SIP Invite	1
SIP Session Expires Timer	86400
Cisco CallManager Service System – QOS Parameters	
System > Service Parameters > Appropriate server > Cisco CallManager (Active) > Advanced Clusterwide Parameters (System - QOS)	
DSCP for Video Calls	34 (100010)
Cisco CallManager Service Enterprise Parameters	
System > Enterprise Parameters	
Advertise G.722 Codec	Enabled
Cisco CallManager Service Cisco IP Voice Media Streaming Application service	
System > Service Parameters > Appropriate server > Cisco IP Voice Media Streaming App (Active)	
MTP Run Flag	False
Supported MOH Codec	G711alaw/G711ulaw, G729 Annex A

Cisco CallManager Service Region configuration	
Menu	**Value**
System > Region Information > Region	
Regions configuration for customer using G.729	<table><tr><td>From To</td><td>HQ</td><td>RS</td><td>WAN</td></tr><tr><td>HQ</td><td>G711</td><td>G729</td><td>G729</td></tr><tr><td>RS</td><td>G729</td><td>G711</td><td>G729</td></tr><tr><td>WAN</td><td>G729</td><td>G729</td><td>G729</td></tr></table>
Regions configuration for customer using G.711	<table><tr><td>From To</td><td>HQ</td><td>RS</td><td>WAN</td></tr><tr><td>HQ</td><td>G711</td><td>G711</td><td>G711</td></tr><tr><td>RS</td><td>G711</td><td>G711</td><td>G711</td></tr><tr><td>WAN</td><td>G711</td><td>G711</td><td>G711</td></tr></table>
Cisco CallManager Service Device Pool Configuration	
System > Device Pool > Add new	
New Device Pool	Device Pool configuration: • The number of Device Pools at least should be the same as the number of site • Every Device Pool should have appropriate Region and Location value **Note:** MOH server requires a separate Device Pool configuration.
Cisco CallManager Service Locations (Call Admission Control)	
System > Location Info> Location > Add new	
New Location	**Warning!** RSVP locations are not supported! Create the necessary locations and configure the bandwidth for each.

Media Resources **Transcoder configuration :** Warning! Hardware MTP resources on IOS Gateway and software MTP resource on CUCM are NOT SUPPORTED. Software MTPs on IOS Gateway are SUPPORTED in BT/BTIP SIP Trunking.	
Menu	**Value**
Media Resources > Transcoder > Add new	
Transcoder Type	Cisco IOS Enhanced Media Termination Point
Device Name	Use the name configured in sccp ccm group in the IOS
Device Pool	Use the appropriate Device Pool
Trusted Rely Point	Unchecked
Media Resources Conference Bridge configuration	
Media Resources > Conference Bridge > Add new	
Conference Bridge Type	Cisco IOS Enhanced Media Termination Point
Device Name	Use the name configured in sccp ccm group in the IOS
Device Pool	Use the appropriate Device Pool
Device Security Mode	Non Secure Conference Bridge
Media Resources Multicast Music on Hold CUCM configuration - Region	
System > Region Information > Region > Add new	
New Region	Please refer to chapter on Region configuration for additional information. With this configuration, all devices in **"MoH Multicast" region will use G.711 as codec for sending RTP packets to devices to all other regions** and also for the "WAN" region where codec G.711 will be used.
Media Resources Multicast Music on Hold CUCM configuration – Device Pool	
System > Device Pool > Add new	
New Device Pool	Choose a name and associate the Region "MoH Multicast" to this new Device Pool.
Media Resources Multicast Music on Hold CUCM configuration - Audio Source Configuration	
Media Resources > Music On Hold Audio Source > Add new	
Play continuously (repeat)	Checked
Allow Multicasting	Checked

Media Resources Multicast Music on Hold CUCM configuration - Multicast MoH server configuration	
Menu	**Value**
Media Resources > Music On Hold Server	
Device Pool	Checked
Enable Multi-cast Audio Sources on this MoH Server	Checked
Base Multi-cast IP Address	239.1.1.1 *(example)*
Base Multi-cast IP Port	16384 *(example)*
Increment Multi-cast on	IP Address
Max Hops (per Audio Source in Selected Audio Sources configuration area)	1
Media Resources Multicast Music on Hold CUCM configuration - Multicast MoH server configuration	
Media Resources > Media Resource Group	
Appropriate Media Resource Group	Check the Use Multicast for MoH Audio checkbox to allow multicast with this resource group.
Media Resources Multicast Music on Hold Router configuration – Audio file	
Frequency	9kHz
Coded with	8bit
Audio mode	Mono
Codec type	CCITT u-law
Media Resources Multicast Music on Hold Router configuration – IOS Commands	
Commands	ccm-manager music-on-hold call-manager-fallback max-conferences 4 ip source-address 10.108.105.254 port 2000 max-ephones 24 max-dn 48 moh TheJourneyAndTheWind.alaw.wav multicast moh 239.1.1.1 port 16384 route 210.72.240.13 10.108.105.254
Media Resources Multicast Music on Hold Media Resource Group Lists configuration	
Media resources	**Warning!** Media Resources, which are not associated with any MRG are available to every device in the cluster by default. Media Resources > Media Resource Group > Add new Resources > Media Resource Group List > Add new
Off-net calling via BT/BTIP Diversion Header manipulation	

Partition	
Menu	**Value**
Call Routing -> Class of Control -> Partition -> Add new	
Name	DIV-HEADER-PT
Off-net calling via BT/BTIP Diversion Header manipulation Called Party Transformation Pattern	
Call Routing -> Transformation -> Transformation Pattern -> Called PartyTransformation Pattern -> Add New	
Pattern	XXXX
Prefix digits	Site Prefix
Off-net calling via BT/BTIP Diversion Header manipulation Calling Search Space	
Call Routing -> Class of Control -> Calling Search Space -> Add New	
Name	DIV-HEADER-CSS
Selected Partitions	DIV-HEADER-PT
Off-net calling via BT/BTIP Basic Configuration Sip Trunk Security Profile	
System > Security > SIP Trunk Security Profile, select “Non Secure SIP Trunk Profile” from SIP Trunk Security Profile List	
Incoming Transport Type	TCP + UDP
Outgoing Transport Type	UDP
Off-net calling via BT/BTIP Basic Configuration SIP Profile	
Device > Device Settings > SIP Profile	
User-Agent and Server header information	Send Unified CM Version Information as User-Agent Header
Version in User Agent and Server Header	Full Build
SIP Rel1XX Options	Send PRACK for 1xx Messages
Early Offer support for voice and video	Mandatory (insert MTP if needed)
Send send-receive SDP in mid-call INVITE	Checked
Ping Interval for In-service and Partially In-service Trunks (seconds)	300
Ping Interval for Out-of-service Trunks (seconds)	5
Version in User Agent and Sever Header	Full build
Session Refresh Method	INVITE or UPDATE

Version in User Agent and Sever Header - inject info about full version of CUCM

Session Refresh Method - since CUCM 10.0 there is additional method – “UPDATE”. “INVITE” should be used by default.

Off-net calling via BT/BTIP

Basic Configuration

SIP Normalization Script

Device > Device Settings > SIP normalization script > Add new

SIP Normalization Script is applied to SIP trunk and is required to adapt the SIP signaling to the form expected by BT/BTIP infrastructure.
The content of the script is given below:

```
-- Orange SIP Normalization Script v11
-- this is normalization script for uc 12.x
M = {}


-- This is called when an INVITE message is sent
function M.outbound INVITE(msg)
    local sdp = msg:getSdp()
    if sdp
    then
       -- remove b=TIAS:
       sdp = sdp:gsub("b=TIAS:%d*\r\n", "")
       -- store the updated sdp in the message object
       msg:setSdp(sdp)
    end
end


--modifying of Server header in 183 messages
function M.outbound 183 INVITE(msg)
-- change 183 to 180 if sdp
 local sdp = msg:getSdp()
 if sdp
 then
  msg:setResponseCode(180, "Ringing")
 end
end

--modifying of Server header in 488 messages
function M.outbound 488 INVITE(msg)
 -- change 488 to 503 if sdp
  msg:setResponseCode(503, "Service Unavailable")
end

--handling of 400 errors
function M.inbound 400 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=27")
 else
  msg:addHeader("Reason", "Q.850; cause=27")
 end
end

--handling of 403 errors
function M.inbound 403 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=2")
 end
end
```

```
--handling of 408 errors
function M.inbound 408 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:removeHeader("Reason")
 end
end

-- handling of 480 errors
function M.inbound 480 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if not reason
 then
  msg:addHeader("Reason", "Q.850; cause=20")
 end
end

--handling of 481 errors
function M.inbound 481 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=27")
 else
  msg:addHeader("Reason", "Q.850; cause=27")
 end
end

--handling of 487 errors
function M.inbound 487 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if not reason
 then
  msg:addHeader("Reason", "Q.850; cause=16")
 end
end

--handling of 488 errors
function M.inbound 488 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if not reason
 then
  msg:addHeader("Reason", "Q.850; cause=127")
 end
end

--handling of 500 errors
function M.inbound 500 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=2")
 else
  msg:addHeader("Reason", "Q.850; cause=2")
 end
end

--handling of 501 errors
function M.inbound 501 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=2")
 else
  msg:addHeader("Reason", "Q.850; cause=2")
 end
end
```

```
--handling of 502 errors
function M.inbound 502 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:removeHeader("Reason")
 end
end

-- handling of 503 errors
function M.inbound 503 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=38")
 else
  msg:addHeader("Reason", "Q.850; cause=38")
 end
end

-- handling of 505 errors
function M.inbound 505 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=38")
 else
  msg:addHeader("Reason", "Q.850; cause=38")
 end
end

-- handling of 513 errors
function M.inbound 513 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=38")
 else
  msg:addHeader("Reason", "Q.850; cause=38")
 end
end

-- addition of PAI header if incoming INVITE includes Privacy
header
function M.inbound INVITE(msg)
 -- get Privacy header
 local privacy = msg:getHeader("Privacy")
 if privacy
 then
  -- get From and Pai
  from = msg:getHeader("From")
  pai = msg:getHeader("P-Asserted-Identity")
  --check if Pai header is not present
  if pai==nil
  then
   -- add Pai header filled with From URI value
   local uri = string.match(from, "(<.+>)")
   msg:addHeader("P-Asserted-Identity", uri)
  end
 end
end

return M
```

Off-net calling via BT/BTIP Basic Configuration SIP Trunk Configuration	
Menu	**Value**
Device > Trunk > Add new	
Device Pool	Choose Device Pool which include Region and Location value
Media Resource Group List	MRGL
Redirecting Diversion Header Delivery - Inbound	Checked
Redirecting Diversion Header Delivery - outbound	Checked
Destination Address	SBC IP Address
SIP Trunk Security Profile	SIP Trunk Security Profile name
SIP Profile	Standard SIP Profile with PRACKs, EO, Send-recv
DTMF Signaling Method	RFC 2833
Normalization Script	SIP Normalization Script name (currently v8)
Enable Trace	Unchecked
Redirecting Party Transformation CSS	DIV-HEADER-CSS
Off-net calling via BT/BTIP Basic Configuration Route Group	
Call Routing > Route/Hunt > Route group > Add new	
Distribution algorithm	Top Down
Selected devices	both SIP trunks to ORACLE/ACMEs
Off-net calling via BT/BTIP Basic Configuration Route List	
Call Routing > Route/Hunt > Route list > Add new	
Selected Groups	Route Group with SIP trunks to BT/BTIP
Off-net calling via BT/BTIP Basic Configuration Route Pattern	
Call Routing > Route/Hunt > Route Pattern > Add new	
Route Pattern	Specific Route Pattern
Gateway/Route List	Route List name
Call Classification	OffNet
Discard Digits	PreDot Trailing#
On-net calling Basic Configuration	
The configuration of such intercluster SIP Trunk is **the same** as the one described for off-net calls except that on trunk between sites there is **no SIP Normalization Script.**	
SME Architecture (ON CUSTOMER DEMAND) Off-net calling via BT/BTIP SIP Trunk Security Profile (at CUCM SME and CUCM)	

Menu	Value
System > Security > SIP Trunk Security Profile > Add new	
Incoming Transport Type	TCP + UDP
Outgoing Transport Type	UDP
SME Architecture **Off-net calling via BT/BTIP** **SIP Trunk Security Profile (at CUCM SME and CUCM)**	
Device > Device Settings > SIP Profile	
User-Agent and Server header information	Send Unified CM Version Information as User-Agent Header
Version in User Agent and Server Header	Full Build
SIP Rel1XX Options	Send PRACK for 1xx Messages
Early Offer support for voice and video calls (insert MTP if needed)	Checked
Send send-receive SDP in mid-call INVITE	Checked
Ping Interval for In-service and Partially In-service Trunks (seconds)	300
Ping Interval for Out-of-service Trunks (seconds)	5
SME Architecture **Off-net calling via BT/BTIP** **SIP Normalization Script (at CUCM SME)**	
Device > Device Settings > SIP normalization script > Add new	

SIP Normalization Script is applied to SIP trunk at CUCM SME and is required to adapt the SIP signaling to the form expected by BT/BTIP infrastructure. Create the script.
The content of the script is given below:

```
-- Orange SIP Normalization Script v11
-- this is normalization script for uc 12.x
M = {}


-- This is called when an INVITE message is sent
function M.outbound INVITE(msg)
    local sdp = msg:getSdp()
    if sdp
    then
       -- remove b=TIAS:
       sdp = sdp:gsub("b=TIAS:%d*\r\n", "")
       -- store the updated sdp in the message object
       msg:setSdp(sdp)
    end
end


--modifying of Server header in 183 messages
function M.outbound 183 INVITE(msg)
-- change 183 to 180 if sdp
 local sdp = msg:getSdp()
 if sdp
 then
  msg:setResponseCode(180, "Ringing")
 end
end

--modifying of Server header in 488 messages
function M.outbound 488 INVITE(msg)
```



```
 -- change 488 to 503 if sdp
  msg:setResponseCode(503, "Service Unavailable")
end

--handling of 400 errors
function M.inbound_400_INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=27")
 else
  msg:addHeader("Reason", "Q.850; cause=27")
 end
end

--handling of 403 errors
function M.inbound_403_INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=2")
 end
end

--handling of 408 errors
function M.inbound_408_INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:removeHeader("Reason")
 end
end

-- handling of 480 errors
function M.inbound_480_INVITE(msg)
 local reason = msg:getHeader("Reason")
 if not reason
 then
  msg:addHeader("Reason", "Q.850; cause=20")
 end
end

--handling of 481 errors
function M.inbound_481_INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=27")
 else
  msg:addHeader("Reason", "Q.850; cause=27")
 end
end

--handling of 487 errors
function M.inbound_487_INVITE(msg)
 local reason = msg:getHeader("Reason")
 if not reason
 then
  msg:addHeader("Reason", "Q.850; cause=16")
 end
end

--handling of 488 errors
function M.inbound_488_INVITE(msg)
 local reason = msg:getHeader("Reason")
 if not reason
 then
  msg:addHeader("Reason", "Q.850; cause=127")
```



```
 end
end

--handling of 500 errors
function M.inbound 500 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=2")
 else
  msg:addHeader("Reason", "Q.850; cause=2")
 end
end

--handling of 501 errors
function M.inbound 501 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=2")
 else
  msg:addHeader("Reason", "Q.850; cause=2")
 end
end

--handling of 502 errors
function M.inbound 502 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:removeHeader("Reason")
 end
end

-- handling of 503 errors
function M.inbound 503 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=38")
 else
  msg:addHeader("Reason", "Q.850; cause=38")
 end
end

-- handling of 505 errors
function M.inbound 505 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=38")
 else
  msg:addHeader("Reason", "Q.850; cause=38")
 end
end

-- handling of 513 errors
function M.inbound 513 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=38")
 else
  msg:addHeader("Reason", "Q.850; cause=38")
 end
end

-- addition of PAI header if incoming INVITE includes Privacy
```

```
header
function M.inbound INVITE(msg)
 -- get Privacy header
 local privacy = msg:getHeader("Privacy")
 if privacy
 then
  -- get From and Pai
  from = msg:getHeader("From")
  pai = msg:getHeader("P-Asserted-Identity")
  --check if Pai header is not present
  if pai==nil
  then
   -- add Pai header filled with From URI value
   local uri = string.match(from, "(<.+>)")
   msg:addHeader("P-Asserted-Identity", uri)
  end
 end
end

return M
```

SME Architecture Off-net calling via BT/BTIP SIP Trunk Configuration to offnet (at CUCM SME)	
Menu	**Value**
Device > Trunk > Add new	
Device Pool	Choose Device Pool which include Region and Location value
Media Resource Group List	None
Redirecting Diversion Header Delivery - Inbound	Checked
Destination Address	SBC IP Address
SIP Trunk Security Profile	SIP Trunk Secure Profile name
SIP Profile	Standard SIP Profile with PRACKs, EO and Send-recv
Normalization Script	SIP Normalization Script name
Enable Trace	Unchecked
SME Architecture Off-net calling via BT/BTIP Route group (at CUCM SME)	
Call Routing > Route/Hunt > Route group > Add new	
Distribution algorithm	Top Down
Selected devices	both SIP trunks to ORACLE/ACMEs
SME Architecture Off-net calling via BT/BTIP Route list (at CUCM SME)	
Call Routing > Route/Hunt > Route list > Add new	
Selected Groups	Route Group with SIP trunks to BT/BTIP
SME Architecture Off-net calling via BT/BTIP Route pattern (at CUCM SME)	
Call Routing > Route/Hunt > Route Pattern > Add new	
Route Pattern	Specific Route Pattern

Gateway/Route List	Route List name
Call Classification	OffNet
Discard Digits	PreDot Trailing#

<table>
<tr><td colspan="2">SME Architecture
On-net calling</td></tr>
<tr><td colspan="2">The configuration of such intercluster SIP Trunk is the same as the one described for off-net calls except for:
Media Resource Group List – should be set to the group containing following resources: conference, transcoder, annuciator (Subscribers), MOH Server (Subscribers), software MTP
SIP Normalization Script should not be added to this trunk
SIP Trunks should be between CUCM of independent site and CUCM SME (there is no direct SIP Trunks between independent sites in SME Architecture – all on-net calls are managed by CUCM SME).</td></tr>
<tr><td colspan="2">Emergency number support for Extension Mobility
Partitions</td></tr>
<tr><td>Menu</td><td>Value</td></tr>
<tr><td>Call Routing > Class of Control > Partition > Add new</td><td>Create a partition for emergency numbers for each site, for example: EN_HQ_PT, EN_RSA_PT, EN_RSB_PT.</td></tr>
<tr><td colspan="2">Route Patterns</td></tr>
<tr><td colspan="2">Call Routing > Route/Hunt > Route Pattern > Add new</td></tr>
<tr><td>Route Partition</td><td>Choose Partition for appropriate Route Pattern</td></tr>
<tr><td>Urgent Priority</td><td>Checked</td></tr>
<tr><td>Calling Party Transform Mask</td><td>Enter valid office attendant phone number (unique for each site)</td></tr>
<tr><td colspan="2">Calling search spaces</td></tr>
<tr><td colspan="2">Call Routing > Class of Control > Calling Search Space > Add new</td></tr>
<tr><td colspan="2">Create a CSS for emergency numbers for each site and another one for non-emergency numbers.

❶ CSS_LINE associated to the line deals with general call right except emergency numbers.
❷ CSS_PHONE associated to the phone deals with emergency calls. This CSS should be unique for each site.</td></tr>
<tr><td colspan="2">Device > Phone > Calling Search Space</td></tr>
<tr><td colspan="2">Associate the calling search spaces for emergency numbers with particular phones (deivces), and calling search spaces for non-emergency numbers with lines.</td></tr>
<tr><td>Device > Phone -> find a phone ->Calling Search Space field</td><td>select the proper CSS</td></tr>
<tr><td>Device > Phone -> find a phone ->select the line on the left menu -> Calling Search Space field</td><td>select the proper CSS</td></tr>
<tr><td colspan="2">Survivable Remote Site Telephony configuration</td></tr>
<tr><td colspan="2">SRST mode is not supported with BT/BTIP infrastructure but with local PSTN gateway configured on CE router</td></tr>
</table>

6 Cisco Unity Connection configuration

Cisco Unified Communication Manager Configuration	
Menu	**Value**
System > Device Pool > Add New	Add new Device pool
Advanced FeaturesVoice Mail > Cisco Voice Mail Port Wizard >	Create a new Cisco Voice Mail Server and add ports to it
Call Routing > Route/Hunt > Line Group	add/configure the Answering Voice Mail Ports to a Line Group
Call Routing > Route/Hunt > Hunt List > Add New	include the Line Group created earlier
Call Routing > Route/Hunt > Hunt Pilot > Add New	include the Hunt List created earlier
Advanced Features > Voice Mail > Message Waiting	add one number for turning MWIs on and one for turning MWIs off
Advanced Features > Voice Mail > Voice Mail Pilot > Add New	Configure the voice mail pilot
Advanced Features > Voice Mail > Voice Mail Profile > Add New	Associate Voice Mail Pilot number created earlier with this profile
Cisco Unity Connection Configuration	
Telephony Integrations > Phone System	Configure the phone system
Phone System Basics > Related Links drop-down box > Add Port Group > Go	Port group configuration
Port Group Basics > Related Links drop-down box > Add Ports > Go	Add and configure required number of ports
Cisco Unity Connection Administration > Telephony Integrations > Port Group	On Search Port Groups page click the display name of the port group that you created with the phone system integration
Port Group Basics page > Edit > Servers >	add backup CUCM servers if needed
BT/BTIP specific parameters	
Telephony Integrations -> Port Group -> choose appropriate -> Edit -> Codec Advertising	change the codec list used for calls to CUC - select G.711 A-law / G.711ulaw/G.722 or G.729 codecs in advertised codecs.
System Setting > General Configuration	Select G.711 a-law, G.711 u-law or G.729 codec as specified for Recording Format parameter

7 Unified Contact Center Express configuration

7.1 Provisioning UCCX (CUCM part)

7.1.1 Adding agents

Unified CM users in Unified CCX are assigned an agent's role when an **agent extension** is associated to the user in the Unified CM User Configuration page. Consequently, this role can only be assigned or removed for the user using Unified CM Administrator's End User configuration web page. These users cannot be assigned or removed in Unified CCX Administration.

Configuring Unified CM users who will be agents in your Unified CCX system:

Step 1 From the **Unified CM Administration** menu bar, choose **User Management > End User**.

Step 2 In the **Controlled Devices** list box below the Device Information section, select the agent's phone device.

Step 3 In the **Primary Extension** field drop-down list and the **IPCC Extension field** drop-down list, choose the required agent extension for this device.

Step 4 Define permissions and roles information:

Groups:

- Standard AXL API Access
- Standard CCM Admin Users
- Standard CTI Allow Call Monitoring
- Standard CTI Allow Call Park Monitoring
- Standard CTI Allow Call Recording
- Standard CTI Allow Calling Number Modification
- Standard CTI Allow Control of All Devices
- Standard CTI Enabled
- Standard Confidential Access Level Users

Roles:

- Standard AXL API Access
- Standard CCM Admin Users
- Standard CTI Allow Call Park Monitoring
- Standard CTI Allow Call Recording

- Standard CTI Allow Calling Number Modification
- Standard CTI Allow Control of All Devices
- Standard CTI Enabled
- Standard CUReporting
- Standard CUReporting Authentication
- Standard Confidential Access Level Users

Step 5 Adding End User to IP phone - End user related to UCCX has to be associated to ip phone profile and ip phone line

7.1.2 Activation and Configuring IP Phone Agent service

Step 1 Activate IP Phone Agent service (URL can be found in CAD administration guide: http:// UCCX_IP_address or FQDN:8082/fippa/#DEVICENAME#): CUCM administration > Device > Device Settings > Phone services

Step 2 Create parameters which will be used to log in IP Phone Agent service: extension, id and password.

Step 3 Subscribe agent phone to this newly created service (Phone > Subscribe services drop-box list)

Step 4 (Optional, if needed) Create an application user named “telecaster” with “telecaster” as the password (or whatever BIPPA user ID and password was specified in the CAD Configuration Setup utility).

Step 5 (Optional, if needed) Assign the telecaster application user to all the IP agent phones

7.1.3 UCCX Application Users on CUCM

When UCCX will be properly configured two Application Users should be created automatically on CUCM:

- RMCM user

  Go to CUCM administration > User Management > Application User > RMCM user

  IP Phone (which will be used as the agent) manually associates with “Device Association” to RMCM user Controlled Device.

- JTAPI user

  Go to CUCM administration > User Management > Application User > JTAPI user

Automatic creation of this user should take place on CUCM **(after proper configuration of UCCX)** and then UCCX CTI ports should appear automatically in the list “Controlled Devices”.

7.2 UCCX part of configuration

7.2.1 Provisioning Call Control Group (CCC)

Provision Unified CM Telephony call control groups (**Subsystems** > **Unified CM Telephony** > **Call Control Group**). They are CTI ports which will be used by UCCX to handle calls

- Define Description
- Define Number of CTI Ports
- Define Name Prefix
- Define Starting Directory Number – unique and not used on CUCM
- Define Device Pool
- (optionally – if needed) Synchronize Cisco JTAPI Client and Unified CM Telephony Data (this creates all necessary CTI devices on CUCM using AXL interface)

Note! Correct behavior - CTI ports should be created and assigned automatically into CCC. CTI ports should be also automatically created and registered on CUCM via AXL integration. If not then perform step 6.

7.2.2 Resources and assignment of skills

Step 1 Check if resources exist – it should exist if former steps of configuration on CUCM and UCCX were performed properly (**Subsystems** > **RmCm** > **Resources**)

Step 2 Create skills (**Subsystems** > **RmCm** > **Skills**)

Step 3 Choose Resource Name and click Add Skill (**Subsystems** > **RmCm** > **Assign Skills**).

Step 4 Assigning skills to agents

Before assigning the skill competence level of the skill should be defined (default is 5)

7.2.3 Configuring Customer Service Queues (CSQ)

Step 1 Creating Contact Service Queues.(**Subsystems** > **RmCm** > **Contact Service Queues**)

Step 2 Define name of CSQ

Step 3 Define type of Resource Pool Selection Model (drop-down list)

Step 4 Click "next" and change default values of parameters of CSQ (if needed), if not just click "update".

Note! Minimum Competence Level shouldn't be higher than formerly defined Competence Level during assigning skills into Resources.

7.2.4 Application and Script configuration

Step 1 Add a new Cisco script application, go to: **Applications > Application Management>Add New** and choose Cisco Script Application:

Step 2 From the Application Type drop-down menu select your script or the standard ICD script **SSCRIPT[icd.aef]** and click "Next"

Step 3 Describe maximum number of sessions (should be "inline" with numbers of CTI ports)

Step 4 Mark checkbox CSQ and enter the name.

Step 5 Define Description

7.2.5 Trigger configuration

Step 1 Add a new Trigger, go to: **Applications > Application Management** and choose application from the list.

Step 2 Choose "Add new trigger"

Step 3 Define Trigger Type and click Next

Step 4 Define **unique** directory number and trigger information (don't forget to assign Call Control Group formerly defined)

Step 5 Perform JTAPI and Data resynchronization (**Subsystems > Cisco Unified CM Telephony**)

Step 6 Check CUCM configuration – CTI Route Point should be automatically created with Trigger number defined on UCCX (**Devices > CTI Route Point**)

Step 7 Check CUCM configuration – this CTI Route Point should be also automatically assigned on JTAPI user (**User Management > Application User**)

8 Cisco Unified Attendant Console configuration

CISCO UNIFIED COMMUNICATION MANAGER	
Device>CTI Route Point>Add New	
Menu	**Value**
User ID	CUDAC
Password	Enter password
Confirm Password	Confirm entered password
User Management > Application User > Add new	
User ID	CUDAC
Password	Enter password
Confirm Password	Confirm entered password
BLF Presence Group	Standard Presence Group
Permissions Information	-Standard Access AXL API -Standard CTI Allow Car Park Monitoring -Standard CTI Allow Calling Number Modification -Standard CTI Allow Control of All Devices -Standard CTI Allow Reception of SRTP Key Material -Standard CTI Enabled -Standard CTI Allow Control of Phones supporting Rollover Mode -Standard CTI Allow Control of Phones supporting Connected Xfer and conf
CISCO UNIFIED ATTENDAND ADMIN	
Menu	**Value**
Installation	• When asked enter the IP address of the machine server is being installed on • If SQL Server Express is already installed enter the SQL Server name, User Name, ale password. If you don't have SQL installed it will be installed automatically • Enter the IP address of CUCM • Enter port number (443) • Enter Application User credentials created before • If certificate security alert from CUCM will be displayed it means connection was successful, accept the certificate • Follow on screen instructions
Database Wizard	• Once installation is completed the database is started, let the wizard to perform necessary configuration, when done, click finish, and restart the computer.
http://<<ip.address.of.Unified.Attendand.Server>>/webadmin/login.aspx	Login to the Attendant Server administration User name: ADMIN Password: CISCO
Engineering > Administrator Management	Let's you change default password

Engineering > Database Management	Parameters for the SQL server, if blank enter IP address of machine where SQL server is installed, specify user name, and password,

Menu	**Value**
Engineering > CUCM connectivity	CUCM parameters, if blank, enter CUCM IP address in name field, port number (443), and user name and password of application user.
Engineering > Database Management	Parameters for the SQL server, if blank enter IP address of machine where SQL server is installed, specify user name, and password of application user
System Configuration > System Device Menagment	
CT Gateway Devices> From	6301 (*example)*
CT Gateway Devices> To	6302 (*example)*
Service Devices> From	6401 (*example)*
Service Devices>To	6402 (*example)*
Park Devices>From	6501 (*example)*
Park Devices>To	6502 (*example)*
System Configuration > System Device Menagment	Synchronize with CUCM (Devices will be added automatically to CUCM)
User Configuration > General Properties	
Minimum internal device digit length	1
Maximum internal device digit length	7
External access number	8
Note! Such configuration is necessary to perform successful delayed transfer. Although etting external access number makes it impossible to perform onnet connections to numbers beginning with 8 (i.e LO BLB) as even though they are seven digits numbers, they are traeted as external numbers. Refer to mantis ticket 2462.	
User Configuration > Queue Management	
Team	Dev1
DDI	6100 (example)
Synchronize with CUCM	Will be automatically added to CUCM as CTI port
User Configuration > Operator Management	
Login Name	OPERATOR1 (example)
Password	Set password
Confirm Password	Confirm password
Associated Queues	Associate queue created in previous step
CISCO UNIFIED ATTENDAND CONSOLE	
Menu	**Value**
Installation	• When asked enter the IP address of Cisco Unified Attendant Server • Select the language for application • Follow on screen instruction until installation I completed
Login	Login with credentials created in previous step

CISCO UNIFIED COMMUNICATION MANAGER	
User Management > Application User > CUDAC	
Controlled Devices	Associate devices added by CUDAC Admin
Device > CTI route point > Route point created by CUDAC Admin	
Media Resource Group List	MRGL_MTP_XCODE

9 CUCM with Cisco Unified Border Element configuration

9.1 General CUBE configuration (flow-through mode by default)

network interface

Note : for two SIP trunks two IP addresses must be configured.

```
interface GigabitEthernet0/0
 description CUBE Voice Interface
 no ip address
 duplex auto
 speed auto
!
interface GigabitEthernet0/0.<INTERFACE>
 description *** CUBE ***
 encapsulation dot1Q <INTERFACE>
 ip address <IP ADDR> <Mask>
```

SNMP Server

```
snmp-server community public RO
snmp-server manager
```

Global settings

```
voice service voip
      mode border-element license capacity [session count]
      allow-connections sip to sip
      sip
          header-passing
          error-passthru
      pass-thru headers unsupp
          no update-callerid
      early-offer forced
          midcall-signaling passthru
          sip-profiles 1
             ip address trusted list
                    ipv4 A.B.C.D    ! primary SBC IP address
                    ipv4 E.F.G.H    ! backup SBC IP address
```

Codecs

For customers using G.711 alaw codec:

```
voice class codec 1
   codec preference 1 g711alaw
```

For customers using G.711 ulaw codec:

```
voice class codec 1
   codec preference 1 g711ulaw
```

For customers using G.729 codec use following configuration:

```
voice class codec 2
   codec preference 1 g729r8
```

SIP User Agent

```
sip-ua
   retry invite 1
```

```
   retry response 2
   retry bye 2
   retry cancel 2
   reason-header override
   connection-reuse
   g729-annexb override
   timers options 1000
```

Support for Privacy and P-Asserted Identity

To enable the privacy settings for the header on a specific dial peer, use the voice-class sip privacy id command in dial peer voice configuration mode:

```
dial-peer voice tag  voip
   voice-class sip privacy id
```

To enable the translation to PAID privacy headers in the outgoing header on a specific dial peer, use the voice-class sip asserted-id pai command in dial peer voice configuration mode:

```
dial-peer voice tag voip
   voice-class sip asserted-id pai
```

9.2 Configuration for a CUCM cluster and two CUBEs

CUBE needs to be configured with physical interface will be configured with a secondary IP address.

```
interface FastEthernet 0/0.<INTERFACE>

   ip address <PRIMARY_IP_ADDR> <Mask>

   ip address <SECONDARY_IP_ADDR> <Mask> secondary
```

CUCM cluster will be configured with 4 different SIP trunks :

- 1st SIP trunk pointing to the primary address of Primary CUBE
- 2nd SIP trunk pointing to the secondary address of Primary CUBE
- 3rd SIP trunk pointing to primary address of Secondary CUBE
- 4th SIP trunk pointing to secondary address of Secondary CUBE

CUCM will be configured with a Route List composed of (at least) 4 Route Groups. Each route group will include SIP trunk to one of CUBE IP Address (Primary or Secondary). On each route group parameters, a specific prefix should be defined (one prefix for each RG). This way the CUBE will be able to route the outgoing calls to the right SBC, depending on this prefix value:

For incoming and outgoing calls for CUCMs side

```
dial-peer voice 1 voip
 description ** to/from site devices - Primary CUCM **
 answer-address <INTERFACE>....
 destination-pattern <INTERFACE>....
 session protocol sipv2
 session target ipv4:<PRIMARY_CUCM_IP_ADDR>
 voice-class codec 1
 voice-class sip options-keepalive up-interval 300 down-interval 300 retry 5
 dtmf-relay rtp-nte
 no vad
!
dial-peer voice 2 voip
 description ** to/from site devices - Backup CUCM **
 preference 1
 answer-address <INTERFACE>....
 destination-pattern <INTERFACE>....
 session protocol sipv2
 session target ipv4:<SECONDARY_CUCM_IP_ADDR>
 voice-class codec 1
 voice-class sip options-keepalive up-interval 300 down-interval 300 retry 5
 dtmf-relay rtp-nte
 no vad

!For outgoing calls (with a prefix to select the target SBC)
dial-peer voice 102 voip
 description ** Outgoing calls - Outbound dial peer - Primary SBC side **
 translation-profile outgoing 113
 huntstop
 destination-pattern 113T
  session protocol sipv2
 session target ipv4:<PRIMARY_SBC_IP_ADDR>
 voice-class codec 1
```

```
 voice-class sip options-keepalive up-interval 300 down-interval 300 retry 5
 voice-class sip send 180 sdp
 dtmf-relay rtp-nte
 no vad
!
dial-peer voice 103 voip
 description ** Outgoing calls - Outbound dial peer - Backup SBC side **
 translation-profile outgoing 114
 huntstop
 destination-pattern 114T
 session protocol sipv2
 session target ipv4:<SECONDARY_SBC_IP_ADDR>
 voice-class codec 1
 voice-class sip options-keepalive up-interval 300 down-interval 300 retry 5
 voice-class sip send 180 sdp
 dtmf-relay rtp-nte
 no vad

!For incoming calls
dial-peer voice 100 voip
 description ** Incoming calls - Inbound dial peer - SBC side **
 answer-address +.T
 session protocol sipv2
 voice-class codec 1
 voice-class sip send 180 sdp
 dtmf-relay rtp-nte
 no vad
!
```

The prefix should be stripped using voice translation rules before sending the call to the infrastructure.

9.3 Configuration for a single CUCM server and one CUBE

CUBE needs to be configured with physical interface will be configured with a secondary IP address.

```
interface FastEthernet 0/0.<INTERFACE>

   ip address <PRIMARY_IP_ADDR> <Mask>

   ip address <SECONDARY_IP_ADDR> <Mask> secondary
```

CUCM will be configured with 2 different SIP trunks :

- 1st SIP trunk pointing to the primary address of the CUBE
- 2nd SIP trunk pointing to the secondary address of the CUBE

CUCM will be configured with a Route List composed of (at least) 2 Route Groups. Each route group will include one of the SIP trunk configured. On each route group parameters, a specific prefix should be defined. This way the CUBE will be able to route the outgoing calls to the right SBC, depending on this prefix value:

```
dial-peer voice 1 voip

   description **CUCMBE**

   answer-address 227....

   destination-pattern 227....

   session target ipv4:<CUCMBE_IP>

   […]


!For outgoing calls (with a prefix to select the target SBC)

dial-peer voice 11 voip

   description ** Outgoing calls - Outbound dial peer - SBC1 side **

   answer-address 227....

   destination-pattern 11T

   session-target <SBC1_IP>

   […]


dial-peer voice 12 voip

   description ** Outgoing calls - Outbound dial peer - SBC2 side **

   answer-address 227....
```



```
   destination-pattern 12T
   session-target <SBC2_IP>
   […]

dial-peer voice 101 voip
 description ** Incoming calls - Inbound dial peer - SBC side **
 answer-address +.T
 voice-class codec 1
 voice-class sip send 180 sdp
 session protocol sipv2
 dtmf-relay rtp-nte
no vad
!
```

9.4 Configuration for a CUCM cluster and one CUBE

CUBE needs to be configured with physical interface will be configured with a secondary IP address.

```
interface FastEthernet 0/0.<INTERFACE>
   ip address <PRIMARY_IP_ADDR> <Mask>
   ip address <SECONDARY_IP_ADDR> <Mask> secondary
```

CUCM cluster will be configured with 2 different SIP trunks :

- 1st SIP trunk pointing to the primary address of the CUBE
- 2nd SIP trunk pointing to the secondary address of the CUBE

CUCM will be configured with a Route List composed of (at least) 2 Route Groups. Each route group will include one of the SIP trunk configured. On each route group parameters, a specific prefix should be defined. This way the CUBE will be able to route the outgoing calls to the right SBC, depending on this prefix value:

For incoming and outgoing calls for CUCMs side

```
dial-peer voice 1 voip
   description **CUCM SUB**
   preference 1
   answer-address 227....
   destination-pattern 227....
   voice-class codec 1
   session target ipv4:<CUCM2_IP>
   […]

dial-peer voice 2 voip
   description **CUCM PUB**
   preference 2
   answer-address 227....
   destination-pattern 227....
   voice-class codec 1
   session target ipv4:<CUCM1_IP>
   […]
```

For outgoing calls (with a prefix to select the target SBC)

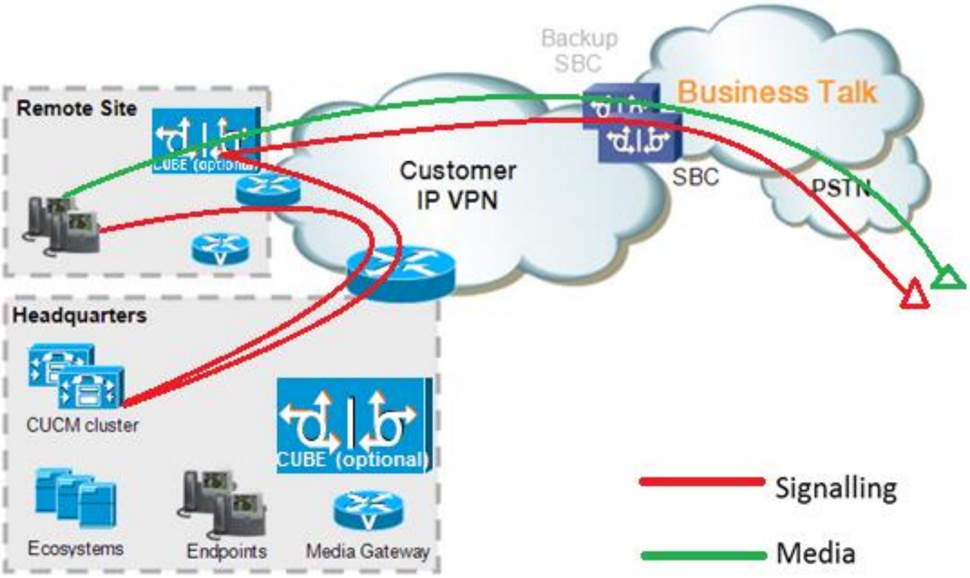
```
dial-peer voice 11 voip
   preference 1
   answer-address 227....
   destination-pattern 11T
   session-target <SBC1_IP>
   […]
dial-peer voice 12 voip
   preference 2
   answer-address 227....
   destination-pattern 12T
   session-target <SBC2_IP>
   […]
```

For incoming calls

```
dial-peer voice 101 voip
 description ** Incoming calls - Inbound dial peer - SBC side **
 answer-address +.T
 voice-class codec 1
 voice-class sip send 180 sdp
 session protocol sipv2
 dtmf-relay rtp-nte
 no vad
!
```

9.5 Design for Local SIP Trunking

For Local SIP Trunking the CUBE configuration remains mostly the same as for the regular configuration. The core differences concerning call routing are decided on CUCM level.



9.5.1 Region configuration

Regions are configured at **System > Region Information > Region.** They need to be associated with proper device pools later.

Codec preference lists can be configured at **System > Region Information > Audio Codec Preference List.** Codec Preference Lists could be assigned to Region configuration, however default option (**Use System Default**) should be set on all regions.

BT/BTIP services currently support only monocodec configuration, i.e. all customer sites need to use the same code. Only one of the 3 following codecs is supported:

- G.729
- G.711 A-law/u-law - CUCM doesn't allow to specify G.711 companding type (A-law or u-law), so simply choose G.711

Note that CUCM does not allow also to differentiate between G.711 and G.722 in Region settings.

Consider the following customer design:

- central site (HQ) with CUCM cluster
- a single remote site (RS) with local CUBE and call processing on HQ

Region	Purpose
HQ	Assigned to devices in the HQ site
RS	Assigned to devices in the Remote Site
WAN	Assigned to SIP trunk to BT/BTIP

Regions configuration example for customer using G.729

G.711/G.722 for intrasite calls and low-bitrate G.729 for calls over the WAN

From To	HQ	RS	WAN
HQ	**G.711/G.722**	G.729	G.729
RS	G.729	**G.711/G.722**	G.729
WAN	G.729	G.729	G.729

Regions configuration example for customer using G.711

G.711 or G.722 used for intrasite calls, for calls over the WAN - G.711.

From To	HQ	RS	WAN
HQ	G.711/G.722	G.711/G.722	G.711
RS	G.711/G.722	G.711/G.722	G.711
WAN	G.711	G.711	G.711

9.5.2 Device Pool configuration

Go to **System > Device Pool** and press **Add new** button.

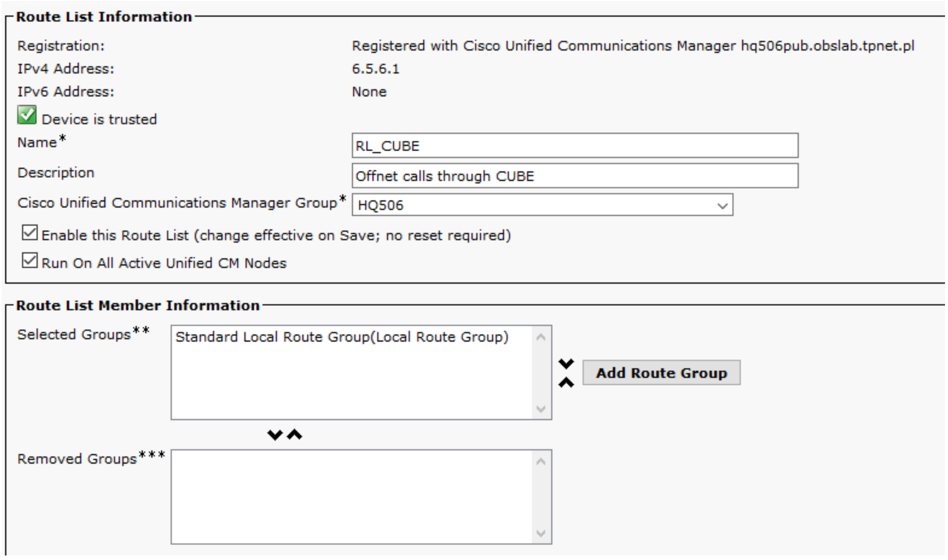
Under Device Pool configuration there are several important parameters:

- The number of Device Pools at least should be the same as the number of sites
- Every Device Pool should has appropriate Region and Location value
- Media Resource Group List need to be add with all resources (annuciator, MOH Server, transcoder, conference, software MTP). See Media Resources section- 2.5).
- **Standard Local Route Group** may be configured in order to enable routing through local CUBE without modifying CSS and partitions. Site-specific Route Group should be set as Standard Local Route Group. If Standard Local Route Group is used, then it should be configured for every device pool depending on the expected trunk to be used. **Note that the Local Route Group used is based on the call originator's device pool in case the call is forwarded.**

Note: MOH server requires a separate Device Pool configuration.

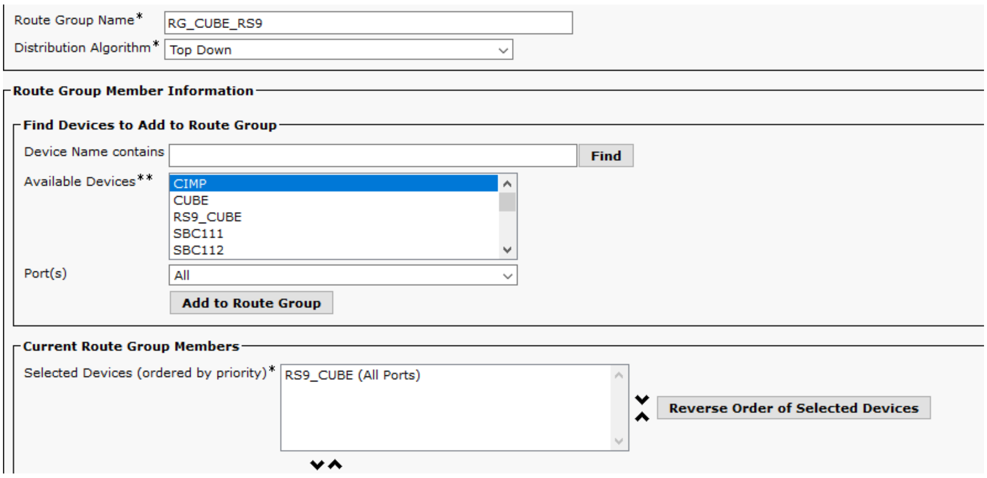
9.5.3 Route List configuration

Standard Local Route Group is configured under the **Route List** used for offnet calls



9.5.4 Route Group Configuration

Route Groups should be configured for each site with trunks used for Offnet calling – either via CUBE or directly towards Orange SBC.



9.5.5 Locations (Call Admission Control)

Go to **System > Location Info > Location** and press **Add new** button.

Warning! RSVP locations are not supported!

For customers using IP VPN to connect all their locations, Static Locations CAC feature in CUCM is well-suited. In such case, **the default Hub_None location with unlimited bandwidth** should be used to represent the IP VPN cloud (no devices should be associated with it). Each site should have a dedicated location to track bandwidth used on its WAN link.

9.5.6 SIP Trunk Configuration

The configuration of SIP Trunks remains standard. Additional SIP Trunks have to be configured toward the Local CUBE. Device Pool used for the trunks toward Local CUBE should be site-specific and contain Standard Local Route Group corresponding to that CUBE. For details on SIP Trunk configuration consult CUCM Configuration Checklist.

10 CUCM with Oracle Session Border Controller configuration

10.1 CUCM configuration

Below is the configuration required on the CUCM side to setup SIP trunk to Oracle SBC. Please note that if some of this configuration has been previously done – for example SIP Profile, it can be reused and there is no need to create separate objects.

Off-net calling via BT/BTIP Diversion Header manipulation Partition	
Menu	**Value**
Call Routing -> Class of Control -> Partition -> Add new	
Name	DIV-HEADER-PT
Off-net calling via BT/BTIP Diversion Header manipulation Called Party Transformation Pattern	
Call Routing -> Transformation -> Transformation Pattern -> Called PartyTransformation Pattern -> Add New	
Pattern	XXXX
Prefix digits	Site Prefix
Off-net calling via BT/BTIP Diversion Header manipulation Calling Search Space	
Call Routing -> Class of Control -> Calling Search Space -> Add New	
Name	DIV-HEADER-CSS
Selected Partitions	DIV-HEADER-PT
Off-net calling via BT/BTIP Basic Configuration Sip Trunk Security Profile	
System > Security > SIP Trunk Security Profile, select "Non Secure SIP Trunk Profile" from SIP Trunk Security Profile List	
Incoming Transport Type	TCP + UDP
Outgoing Transport Type	UDP
Off-net calling via BT/BTIP Basic Configuration SIP Profile	
Device > Device Settings > SIP Profile	
User-Agent and Server header information	Send Unified CM Version Information as User-Agent Header
Version in User Agent and Server Header	Full Build
SIP Rel1XX Options	Send PRACK for 1xx Messages
Early Offer support for voice and video	Mandatory (insert MTP if needed)
Send send-receive SDP in mid-call INVITE	Checked
Ping Interval for In-service and Partially In-service Trunks (seconds)	300

Ping Interval for Out-of-service Trunks (seconds)	5
Version in User Agent and Sever Header	Full build
Session Refresh Method	INVITE or UPDATE

Version in User Agent and Sever Header - inject info about full version of CUCM
Session Refresh Method - since CUCM 10.0 there is additional method – “UPDATE”. “INVITE” should be used by default.

Off-net calling via BT/BTIP

Basic Configuration

SIP Normalization Script

Device > Device Settings > SIP normalization script > Add new

SIP Normalization Script is applied to SIP trunk and is required to adapt the SIP signaling to the form expected by BT/BTIP infrastructure.
The content of the script is given below:

```
-- Orange SIP Normalization Script v11
-- this is normalization script for uc 12.x
M = {}


-- This is called when an INVITE message is sent
function M.outbound INVITE(msg)
    local sdp = msg:getSdp()
    if sdp
    then
       -- remove b=TIAS:
       sdp = sdp:gsub("b=TIAS:%d*\r\n", "")
       -- store the updated sdp in the message object
       msg:setSdp(sdp)
    end
end


--modifying of Server header in 183 messages
function M.outbound 183 INVITE(msg)
-- change 183 to 180 if sdp
 local sdp = msg:getSdp()
 if sdp
 then
  msg:setResponseCode(180, "Ringing")
 end
end

--modifying of Server header in 488 messages
function M.outbound 488 INVITE(msg)
 -- change 488 to 503 if sdp
  msg:setResponseCode(503, "Service Unavailable")
end

--handling of 400 errors
function M.inbound 400 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=27")
```

```
 else
  msg:addHeader("Reason", "Q.850; cause=27")
 end
end

--handling of 403 errors
function M.inbound 403 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=2")
 end
end

--handling of 408 errors
function M.inbound 408 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:removeHeader("Reason")
 end
end

-- handling of 480 errors
function M.inbound 480 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if not reason
 then
  msg:addHeader("Reason", "Q.850; cause=20")
 end
end

--handling of 481 errors
function M.inbound 481 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=27")
 else
  msg:addHeader("Reason", "Q.850; cause=27")
 end
end

--handling of 487 errors
function M.inbound 487 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if not reason
 then
  msg:addHeader("Reason", "Q.850; cause=16")
 end
end

--handling of 488 errors
function M.inbound 488 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if not reason
 then
  msg:addHeader("Reason", "Q.850; cause=127")
 end
end

--handling of 500 errors
function M.inbound 500 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=2")
 else
```



```
  msg:addHeader("Reason", "Q.850; cause=2")
 end
end

--handling of 501 errors
function M.inbound 501 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=2")
 else
  msg:addHeader("Reason", "Q.850; cause=2")
 end
end

--handling of 502 errors
function M.inbound 502 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:removeHeader("Reason")
 end
end

-- handling of 503 errors
function M.inbound 503 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=38")
 else
  msg:addHeader("Reason", "Q.850; cause=38")
 end
end

-- handling of 505 errors
function M.inbound 505 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=38")
 else
  msg:addHeader("Reason", "Q.850; cause=38")
 end
end

-- handling of 513 errors
function M.inbound 513 INVITE(msg)
 local reason = msg:getHeader("Reason")
 if reason
 then
  msg:modifyHeader("Reason", "Q.850; cause=38")
 else
  msg:addHeader("Reason", "Q.850; cause=38")
 end
end

-- addition of PAI header if incoming INVITE includes Privacy
header
function M.inbound INVITE(msg)
 -- get Privacy header
 local privacy = msg:getHeader("Privacy")
 if privacy
 then
  -- get From and Pai
  from = msg:getHeader("From")
  pai = msg:getHeader("P-Asserted-Identity")
  --check if Pai header is not present
```

```
  if pai==nil
  then
   -- add Pai header filled with From URI value
   local uri = string.match(from, "(<.+>)")
   msg:addHeader("P-Asserted-Identity", uri)
  end
 end
end

return M
```

Off-net calling via BT/BTIP Basic Configuration SIP Trunk Configuration	
Menu	**Value**
Device > Trunk > Add new	
Device Pool	Choose Device Pool which include Region and Location value
Media Resource Group List	MRGL
Redirecting Diversion Header Delivery - Inbound	Checked
Redirecting Diversion Header Delivery - outbound	Checked
Destination Address	Oracle SBC IP Address
SIP Trunk Security Profile	SIP Trunk Security Profile name
SIP Profile	Standard SIP Profile with PRACKs, EO, Send-recv
DTMF Signaling Method	RFC 2833
Normalization Script	SIP Normalization Script name (currently v11)
Enable Trace	Unchecked
Redirecting Party Transformation CSS	DIV-HEADER-CSS
Media Termination Point Required	Checked
Off-net calling via BT/BTIP Basic Configuration Route Group	
Call Routing > Route/Hunt > Route group > Add new	
Distribution algorithm	Top Down
Selected devices	SIP trunk to ORACLE SBC
Off-net calling via BT/BTIP Basic Configuration Route List	
Call Routing > Route/Hunt > Route list > Add new	
Selected Groups	Route Group with SIP trunk to Oracle SBC
Off-net calling via BT/BTIP Basic Configuration Route Pattern	
Call Routing > Route/Hunt > Route Pattern > Add new	
Route Pattern	Specific Route Pattern
Gateway/Route List	Route List name
Call Classification	OffNet

Discard Digits	PreDot Trailing#

10.2 Oracle SBC configuration

For detailed information regarding Oracle ESBC configuration, please refer to Annex A and dedicated VISIT SIP Configuration Guideline for Oracle ESBC 8.2.

10.2.1 Oracle SBC information required for CUCM interconnection

The pieces of information needed to create a new customer on the SBC are the following ones:

Customer related data		
Code	Content	Example
<VENDOR_IPBX>	Unique identifier of the CISCO CUCM IPBX in the SBC. This field must follow 7 alphabetical characters format.	CISCO
<VLAN_ID>	It corresponds to the VLAN tag allocated to the customer. This field must follow 3 digits format.	110
NOMINAL SBC related data		
<ESBC_SOUTH_NOMINAL_GW>	IP address of the gateway in front of the nominal SBC (PE router) on access side.	138.132.169.1
<ESBC_SOUTH_NOMINAL_IP>	IP address of the nominal SBC South Side on the interconnection network. Cisco IPBXs will send/receive their signaling and media traffic to/from this IP address (on default port 5060 for signaling). This SBC IP address is located in /29 network provided by the customer. It is used to interconnect the nominal SBC on the customer private network.	138.132.169.2
BACKUP SBC related data		
<ESBC_SOUTH_BACKUP_GW>	IP address of the gateway in front of the backup SBC (PE router) on access side.	138.132.179.1
<ESBC_SOUTH_BACKUP_IP>	IP address of the backup SBC SBC South Side on the interconnection network. Cisco IPBXs will send/receive their signaling and media traffic to/from this IP address (on default port 5060 for signaling). This SBC IP address is located in /29 network provided by the customer. It is used to interconnect the backup SBC on the customer private network.	138.132.179.2

10.2.2 Oracle SBC information required for a new IPBX

This chapter specifies which IP addresses need to be indicated in the session agents and the distribution of the session agents in the session agent groups.
The information indicated in the document will help you to fill in the table here after.

The pieces of information needed to create a new IPBX on the e SBC are the following ones:

IPBX related data		
Code	Content	Example
<PBX type>	PBX type, version and configuration. Information needed to define which SA and SAG need to be created, and if specific profile is required.	Cisco CUCM 12.0

<SIP_PROFILE>	This identifier is used to differentiate several SIP profiles. It depends on the type of IPBX (Vendor & version). Specific SBC configuration is linked to each profile, each one corresponding to a Prod+ template. The profile follows 2 digits format. Values: 00: Default profile is number 00 05: Cisco CUCM	05
<Number of Elements for nominal IPBX>	Number of signaling entities to be declared as SA and included in the nominal SAG.	2
<Number of Elements for backup IPBX>.	Number of signaling entities to be declared as SA and included in the backup SAG.	2
<IPBX_NOMINAL_SA1_IP> to <IPBX_NOMINAL_SAn_IP>	IP addresses of the IPBX signaling entities. These entities belong to nominal session agent group.	6.5.6.1 6.5.6.2
<IPBX_BACKUP_SA1_IP> to <IPBX_BACKUP_SAn_IP>	IP addresses of the IPBX signaling entities. These entities belong to backup session agent group.	6.5.6.1 6.5.6.2
<SA_X>	It is a 2 digits number representing the element number within the nominal IPBX. X is varying from 1 to < Number of Elements for nominal IPBX>	01
<SA_Y>	It is a 2 digits number representing the element number within the backup IPBX. Y is varying from 1 to < Number of Elements for backup IPBX>.	01

10.2.3 Information required for BTIP / Btalk SIP Infrastructure

This chapter specifies which IP addresses need to be indicated in the session agents and the distribution of the session agents in the session agent groups.
The information indicated in the document will help you to fill in the table here after.

The pieces of information needed to create a new IPBX on the e SBC are the following ones:

IPBX related data		
Code	Content	Example
<BT_NOMINAL_SA_IP>	IP addresses of the BT/BTIP signaling entities. These entities belong to nominal session agent group.	172.22.246.33 X.X.X.X.
<BT_BACKUP_SA_IP>	IP addresses of the BT/BTIP signaling entities. These entities belong to backup session agent group.	172.22.246.73 X.X.X.X
<SA_X>	It is a 2 digits number representing the element number within the nominal C-SBC. X is varying from 1 to < Number of Elements for nominal ESBC>	01
<SA_Y>	It is a 2 digits number representing the element number within the backup C-SBC. Y is varying from 1 to < Number of Elements for backup ESBC>.	01

10.2.4 SBC Object naming convention

Based on previous information, the following table presents identifiers that will be created in SBC configuration. These unique identifiers are mandatory to configure the SBC. The rules presented below are valid for both Nominal and Backup A-SBC.

SBC OBJECTS	
Name	Description
Customer identifier	Unique identifier of the customer within the SBC on the access part. It is used to configure the name of the access parent realm. Rule is: ACC_<VLAN_ID>_<IPBX_VENDOR>

Nominal IPBX identifier	Unique identifier of the Nominal IPBX within the SBC. It is used to configure the nominal Session-Agent-Group. It is proposed to used the SIP profile, VLAN Id and the T1T7 parameters to configure it. Rule is: N_<VLAN_ID>_<IPBX_VENDOR>_SIP_PROFILE>
Backup IPBX identifier	Unique identifier of the Backup IPBX within the SBC. It is used to configure the backup Session-Agent-Group. It is proposed to used the SIP profile, VLAN Id and the T1T7 parameters to configure it. Rule is: B_<VLAN_ID>_<IPBX_VENDOR>_<SIP_PROFILE>
Element [X] identifier for the Nominal IPBX	Unique identifier of the Element X of the Nominal IPBX within SBC. It is used to configure the nominal Session-Agent that will be included in the nominal Session-Agent-Group. It is proposed to used the VLAN Id and the T1T7 parameters to configure it. Rule is: N-<VLAN_ID>-<IPBX_VENDOR>-<SA_X> Note that underscores are not allowed in hostnames of Session-Agents. Hence, hyphens are used instead.
Element [Y] identifier for the Backup IPBX	Unique identifier of the Element Y of the Backup IPBX within SBC. It is used to configure the backup Session-Agent that will be included in the backup Session-Agent-Group. It is proposed to used the VLAN Id and the T1T7 parameters to configure it. Rule is: B-<VLAN_ID>-<IPBX_VENDOR>-<SA_Y>

Maximum size of any identifier is not larger than 24.

10.2.5 Certificate

In “TLS/ Secured SIP Trunking” context, following requirements regarding Certificate configuration:

- Certificate of the certification authority (CA), signing the ESBC certificate(format X.509 Base64)
- 1 cyphered file containing both the private key and the public certificate per domain used on the ESBC, signed by a public trusted Certificate Authority to be known, aka such as Digicert CA which Orange has chosen as CA provider
- Certificate of the trusted certificate authority, and of each sub-authority having signed the above certificate (format X.509 Base64)

10.2.6 Licenses & ESBC entitlement setup

Configuration which will enable the support of the new license model based on provisioned entitlements are not covered in this configuration Guideline such as :

- adding session capacity (based on purchased capacity)
- adding new features (based on purchased license as well). Typically the case for enabling SRTP session.

11 Expressway

11.1 Architecture overview

Server components description

- Expressway Control server (Expressway C): This server is deployed on the same Datacenter LAN than UC applications inside the datacenter. The Expressway C is a SIP proxy and communication Gateway for CUCM.

- Expressway Edge server (Expressway E): This server is deployed on a DMZ inside the datacenter. The Expressway E is a SIP Proxy for devices which are located outside the internal network.

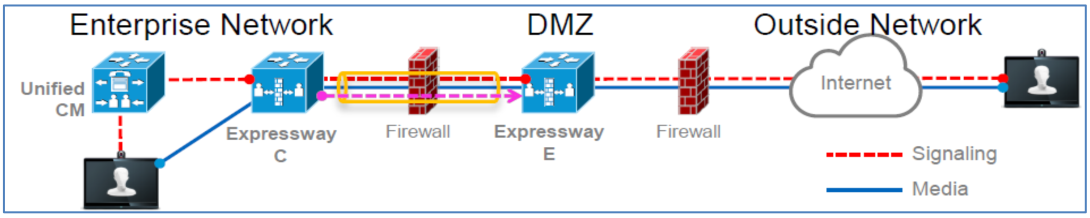
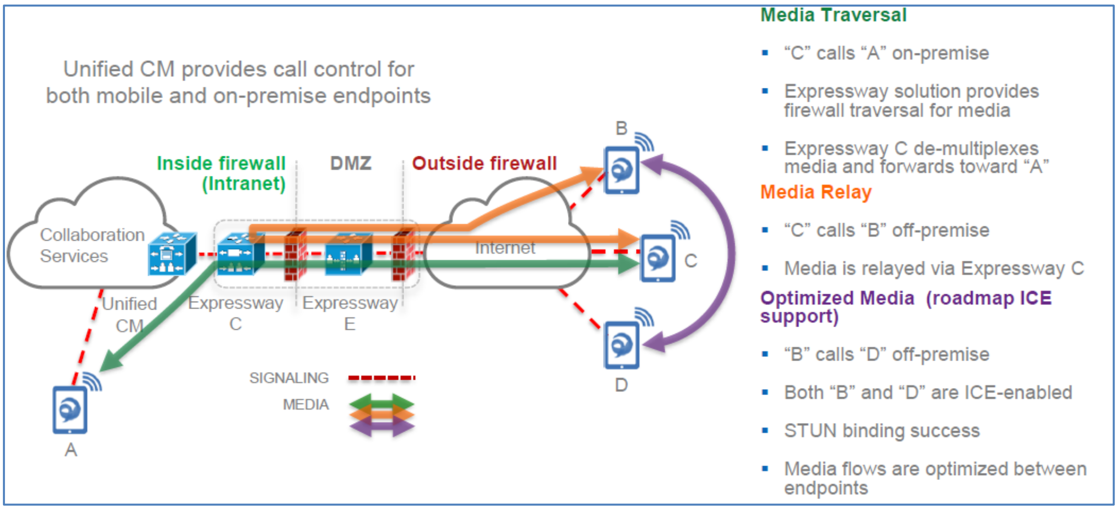


Figure 12-1 – Expressway Firewall Traversal Basics

1. Expressway E is the traversal server installed in DMZ. Expressway C is the traversal client installed inside the enterprise network.
2. Expressway C initiates traversal connections outbound through the firewall to specific ports on Expressway E with secure login credentials.
3. Once the connection has been established, Expressway C sends keep-alive packets to Expressway E to maintain the connection.
4. When Expressway E receives an incoming call, it issues an incoming call request to Expressway C.
5. Expressway C then routes the call to Unified CM to reach the called user or endpoint.
6. The call is established and media traverses the firewall securely over an existing traversal connection.

11.2 Call Flows

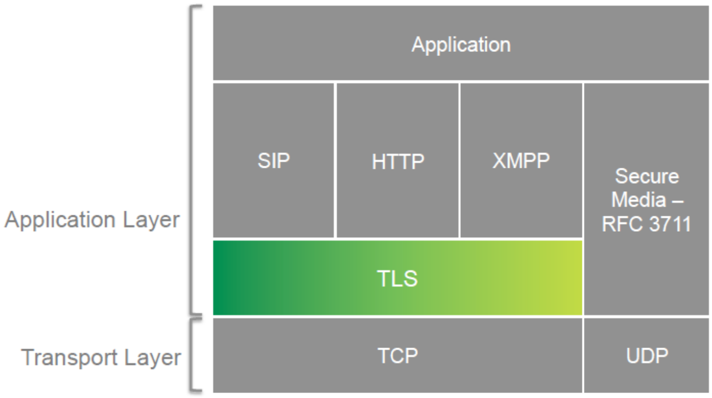
All mobile traffic from the internet is seen with the private Expressway-C IP address on the Customer Network.
All Mobile traffic from the customer network will be seen with the Expressway-E public IP address on the Internet.
The couple Expressway-C and Expressway-E can be seen as a proxy for call flows.
Within VISIT scope, the traffic from the internet would pass through Expressway-C and Expressway-E, through customer managed Call Manager cluster and routed further towards SIP trunk to BT/BTIP infrastructure.



11.3 Endpoint Authentication & Encryption

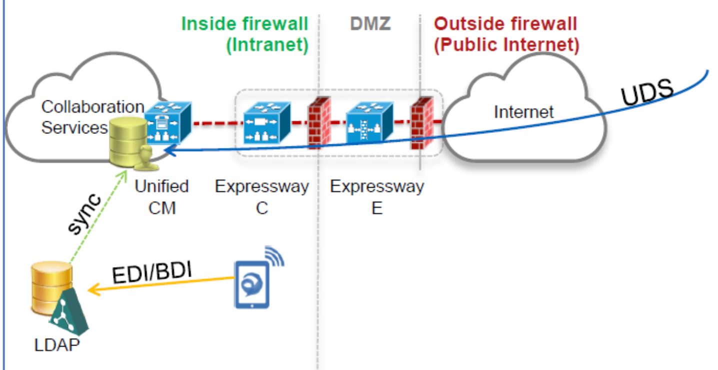
11.3.1 Authentication

Expressway use TLS which is a protocol on top of TCP layer:



11.3.2 Directory integration

Remote Jabber clients will have access to directory look-up services. Cisco Expressway uses the UDS integration model. UDS model relies on the CUCM database for directory search and phone number lookup



- Jabber allows for multiple contact source integrations
- LDAP Directory sync provides corporate directory to Unified CM
- User Data Services (UDS) is a Unified CM RESTful API allowing for contact search, among other things
- All Jabber clients connecting via Expressway will use UDS for contact search
- **Jabber clients deployed on-premise** will use **LDAP** for directory search
- Jabber clients will automatically use UDS for directory search when connecting via Expressway
- The entire corporate directory needs to be sync'd on every Unified CM cluster for best contact search experience

11.3.3 Telephony features

Cisco Jabber endpoints can be deployed using a model in which Cisco Unified Presence and Cisco Unified Communications Manager provide client configuration, instant messaging and presence, user and device management while Microsoft Active Directory provides user lookup/directory search services.

NOTE: Within VISIT scope, all currently supported features continue to function with Expressway infrastructure deployed.

Restriction: An issue has been identified that causes Jabber users registered through Expressway to not fall back to backup server in case nominal server is down.

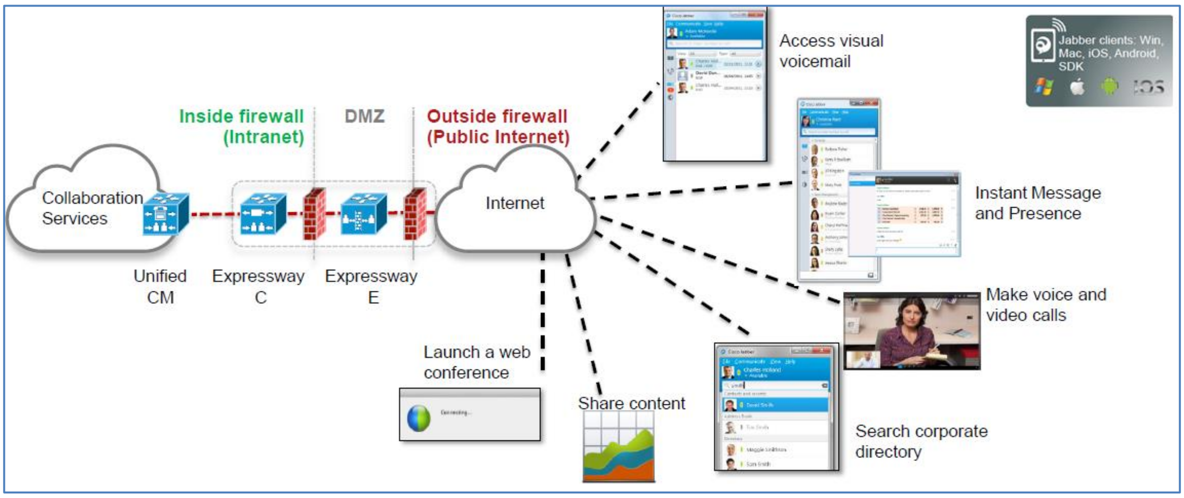
11.4 CUCM configuration update

Mobile and remote access provided by Expressway is, for most part, transparent to Cisco Unified Communications Manager. There is:

- No requirement to build a SIP trunk on CUCM to Expressway C or E,
- No requirement to make dial plan changes ,
- No remote access policy mechanism to limit edge access to certain Jabber users or devices.

Remote Jabber clients or Tele-Presence Endpoints registering to CUCM through Expressway will appear to CUCM as Expressway C IP address (opportunity for CUCM Device Mobility feature usage).

11.5 Expressway specific configuration



This solution allows Jabber clients to securely traverse the enterprise firewall and access collaboration services deployed on the enterprise network. Remote Jabber clients will have access to voice/video, instant messaging and presence, visual voicemail, and directory look-up services.

This section describes the configuration steps required on the Expressway-C.

Configuring DNS and NTP settings

Check and configure the basic system settings on Expressway:

1. Ensure that System host name and Domain name are specified (System > DNS).
2. Ensure that local DNS servers are specified (System > DNS).
3. Ensure that all Expressway systems are synchronized to a reliable NTP service (System > Time). Use an Authentication method in accordance with your local policy.

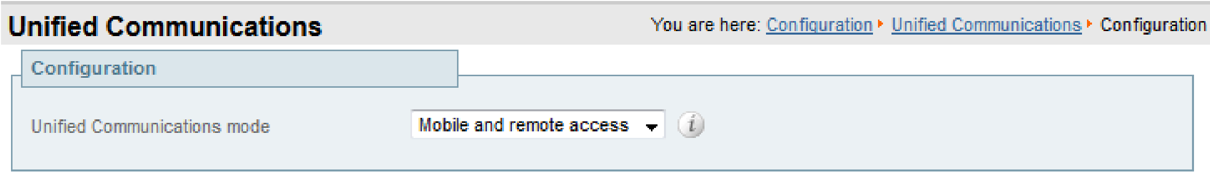
If you have a cluster of Expressways you must do this for every peer.

Configuring the Expressway-C for Unified Communications

To enable mobile and remote access functionality:

1. Go to Configuration > Unified Communications > Configuration.
2. Set Unified Communications mode to Mobile and remote access.

3. Click Save.



Mobile and Remote Access

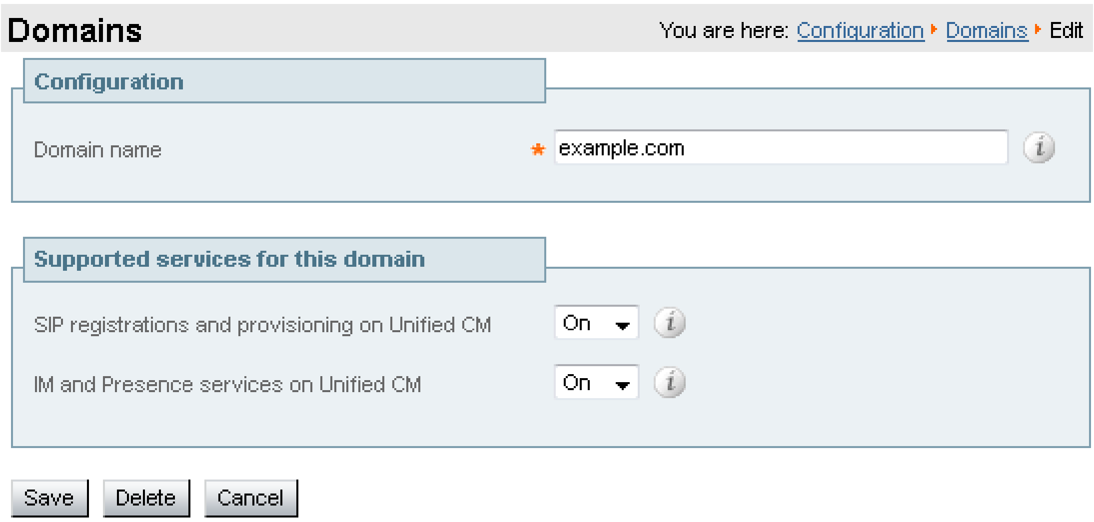
Note that you must select *Mobile and remote access* before you can configure the relevant domains and traversal zones.

Configuring the domains to route to Unified CM

You must configure the domains for which registration, call control, provisioning, messaging and presence services are to be routed to Unified CM.

1. On Expressway-C, go to Configuration > Domains.
2. Select the domains (or create a new domain, if not already configured) for which services are to be routed to Unified CM.
3. For each domain, turn On the services for that domain that Expressway is to support. The available services are:
    - SIP registrations and provisioning on Unified CM: endpoint registration, call control and provisioning for this SIP domain is serviced by Unified CM. The Expressway acts as a Unified Communications gateway to provide secure firewall traversal and line-side support for Unified CM registrations.
    - IM and Presence services on Unified CM: instant messaging and presence services for this SIP domain are provided by the Unified CM IM and Presence service.

Turn On all of the applicable services for each domain.



Discovering IM&P and Unified CM servers

The Expressway-C must be configured with the address details of the IM&P servers and Unified CM servers that are to provide registration, call control, provisioning, messaging and presence services. Note that IM&P server configuration is not required in the hybrid deployment model.

Uploading the IM&P / Unified CM tomcat certificate to the Expressway-C trusted CA list

If you intend to have **TLS verify mode** set to *On* (the default and recommended setting) when discovering the IM&P and Unified CM servers, the Expressway-C must be configured to trust the tomcat certificate presented by those IM&P and Unified CM servers.

1. Determine the relevant CA certificates to upload:
    - If the servers are using self-signed certificates, the Expressway-C's trusted CA list must include a copy of the tomcat certificate from every IM&P / Unified CM server.
    - If the servers are using CA-signed certificates, the Expressway-C's trusted CA list must include the root CA of the issuer of the tomcat certificates.
2. Upload the trusted Certificate Authority (CA) certificates to the Expressway-C (Maintenance > Security certificates > Trusted CA certificate).
3. Restart the Expressway-C for the new trusted CA certificates to take effect (Maintenance > Restart options).

Configuring IM&P servers

To configure the IM&P servers used for remote access:

1. On Expressway-C, go to Configuration > Unified Communications > IM and Presence servers. The resulting page displays any existing servers that have been configured.
2. Add the details of an IM&P publisher:
    a. Click New.
    b. Enter the IM and Presence publisher address and the Username and Password credentials required to access the server. The address can be specified as an FQDN or as an IP address; we recommend using FQDNs when TLS verify mode is On.
    Note that these credentials are stored permanently in the Expressway database. The IM&P user must have the Standard AXL API Access role.
    c. We recommend leaving TLS verify mode set to On to ensure Expressway verifies the tomcat certificate presented by the IM&P server for XMPP-related communications.
        - If the IM&P server is using self-signed certificates, the Expressway-C's trusted CA list must include a copy of the tomcat certificate from every IM&P server.
        - If the IM&P server is using CA-signed certificates, the Expressway-C's trusted CA list must include the root CA of the issuer of the tomcat certificate.
    d. Click Add address.
    The system then attempts to contact the publisher and retrieve details of its associated nodes.

IM&P Servers

Note that the status of the IM&P server will show as Inactive until a valid traversal zone connection between the Expressway-C and the Expressway-E has been established (this is configured later in this process).

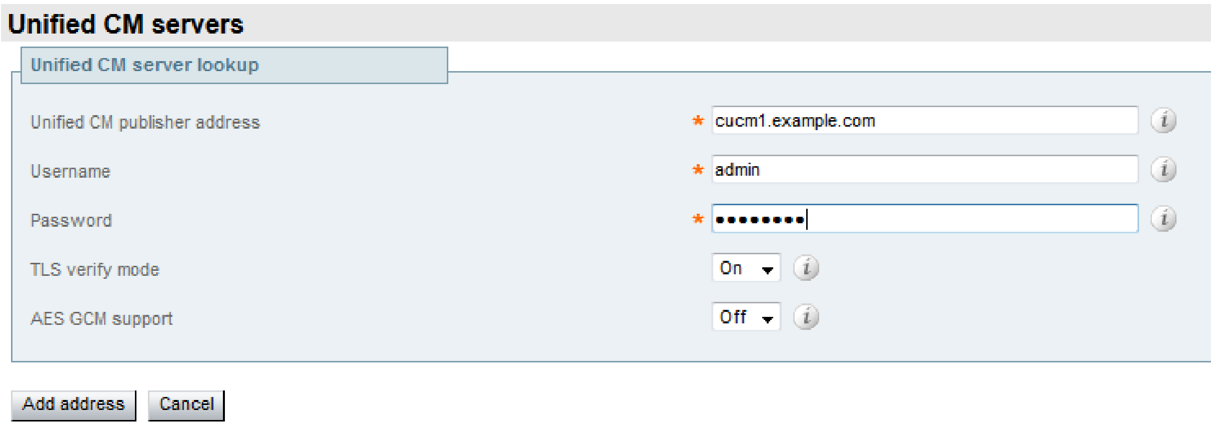
3. Repeat for every IM&P cluster.

After configuring multiple publisher addresses, you can click Refresh servers to refresh the details of the nodes associated with selected addresses.

Configuring Unified CM servers

To configure the Unified CM servers used for remote access:

1. On Expressway-C, go to Configuration > Unified Communications > Unified CM servers. The resulting page displays any existing servers that have been configured.
2. Add the details of a Unified CM publisher:



12 Fax

12.1 Configuration for BT/BTIP SIP trunking

The following guide is an addition to standard SIP Trunk configuration between CUCM and VG. For more details about configuration details and steps to be done on CUCM please refer to following document:

- BTIP/BT SIP System Release 12.0 IOS Voice Gateway Configuration Guide).

12.1.1 T.38 global settings

Below configuration commands are issued under voice gateway's **fax** subcommand menu.

```
voice service voip
   fax
       fax protocol t38 ls-redundancy 4 hs-redundancy 1 fallback none
```

Command	Explanation
fax protocol *protocol* ls-redundancy *value* hs-redundancy *value* fallback *type*	Choice of global fax protocol with assingment of proprer redundacy values and fallbak type

12.1.2 Codec configuration

Below configuration commands are issued under voice gateway's **voice class codec** ***tag*** subcommand menu.

```
voice class codec 1
   codec preference 1 g711alaw
   codec preference 2 g729r8
   codec preference 3 g711ulaw
```

Command	Explanation
codec preference *number codec*	*number* sets priority order (1 = Highest) *codec* sets specific codec format

12.1.3 Example of VoIP dial-peer configuration

Below configuration commands are issued under voice gateway's **dial-peer voice** subcommand menu.

```
dial-peer voice 1 voip
 preference 1
 destination-pattern .T
 session protocol sipv2
 session target ipv4:6.3.9.1
 incoming called-number .
 voice-class codec 1
 dtmf-relay rtp-nte
 fax-relay sg3-to-g3
 fax rate 14400 bytes 72
 fax nsf 000000
```

Command	Explanation
fax-relay *type*	Choice of preffered SG3 to G3 fallback method (CM blocking in TDM to IP direction)
fax rate *speed* bytes *payload*	Specifies desired speed of fax page transmission and payload
fax nsf *000000*	Specifies the fax not to use “non standard facilities”

12.1.4 POTS dial-peer

Below configuration commands are issued under voice gateway’s **dial-peer voice** subcommand menu.

```
dial-peer voice 102 pots
   description fax
   destination-pattern 39001
   progress_ind alert strip
   port 0/0/0
   forward-digits all
```

Command	Explanation
description *description*	Adds a description to the dial peer.
destination-pattern pattern	Sets the destination pattern.
progress_ind alert strip	Allows the media gateway to send a 180 ringing instead of 183 progress SDP. Used to fix RBT generation issues.
port *voice-port*	Specifies the voice port, which should be used to route the call
forward-digits all	Specifies that all digits will be forwarded to the endpoint connected to FXS port.

12.1.5 CUCM Configuration

Below are the steps necessary in order to configure a connection to a VG in a non-standard architecture.

SIP Trunk configuration (*Device -> Trunk*):

Parameter	Value
Trunk Type	SIP Trunk
Device Protocol	SIP
Trunk Service Type	Default
Device Name	TRK-<Site>-<VG Name>
Description	SIP trunk to specific VG
Device Pool	DPO-SIPTRK-<Site>
Location	LOC-<Site>
Call Classification	OnNet
Media Resource Group List	< None >
SRTP Allowed	Not Checked
Run On All Active Unified CM Nodes	Not Checked
Call Routing Information – Inbound Calls	
Significant digits	All

Calling Search Space	CSS-VCGVLG- Enhanced-<CTY><Site>
Redirecting Diversion Header Delivery - Inbound	Checked
Call Routing Information – Outbound Calls	
Calling Party selection	Originator
Redirecting Diversion Header Delivery – Outbound	Checked
Use Device Pool Called Party Transformation CSS	Checked
Use Device Pool Calling Party Transformation CSS	Checked
SIP Information	
Destination Address	<IP address of VG>
Destination Address is an SRV	Not Checked
Destination Port	5060
Rerouting Calling Search Space	CSS-VCGVLG- Enhanced-<CTY><Site>
Out-of-Dialog Refer Calling Search Space	CSS-VCGVLG- Enhanced-<CTY><Site>
SIP Trunk Secure Profile	SIPT-GW
SIP Profile	SIPP-GW
DTMF Signaling Method	RFC 2833

Route Group configuration (*Call Routing -> Route/Hunt -> Route Group*):

Route Group Name	ROG-<Site>-<VG Name>
Distribution Algorithm	TopDown
Selected Devices	TRK-<Site>-<VG Name>

Route List configuration (*Call Routing -> Route/Hunt -> Route List*):

Name	ROL-<Site>-<VG Name>
Description	RL for specific OnNet range to VG SIP controlled device
CUCM Group	CMG01
Enable this Route List	Checked
Run On All Active Unified CM Nodes	Checked
Selected Groups	ROG-<Site>-<VG Name>

Route Pattern configuration (*Call Routing -> Route/Hunt -> Route Pattern*):

Route Pattern	Private Directory Number toward Fax
Route Partition	PAR-Shared
Description	Route Pattern to Fax
Route Class	Default
Gateway / Route List	ROL-<Site>-<VG Name>
Route option	Route this pattern
Call Classification	OnNet
Urgent Priority	Not Checked
Use Calling Party's EPNM	Checked

Translation Pattern configuration (*Call Routing -> Translation Pattern*):

Translation Pattern	Private range toward Fax range i.e. \+4822538.XXXX
Partition	PAR-ForcedOnNet
Description	OnNet calls to VG Fax
Calling Search Space	CSS-AutoAnswer
Route option	Route this pattern
Urgent Priority	Not Checked
Called Party Transformation	
Discard option	Predot
Prefix	InterSite Prefix + SLC (Site Location Code)

12.1.6 CUBE Configuration

In order to enable CUBE IP2IP gateway functionality, following command has to be entered:

```
voice service voip
      mode border-element license capacity [session count]
      allow-connections sip to sip
      sip
          header-passing
          error-passthru
          no update-callerid
        early-offer forced
          midcall-signaling passthru
          sip-profiles 1
             ip address trusted list
                    ipv4 A.B.C.D   ! primary SBC IP address
                    ipv4 E.F.G.H   ! backup SBC IP address
```

Explanation

Command	Description
mode border-element license capacity [session count]	[session count] – indicate the session count based on the license purchased for CUBE
allow-connections sip to sip	Allow IP2IP connections between two SIP call legs
header-passing error-passthru	Error messages are passed through CUBE (SIP error transparency)
no update-callerid	Transparency regarding Caller ID
early-offer forced	Enables SIP Delayed-Offer to Early-Offer globally
midcall-signaling passthru	Passes SIP messages from one IP leg to another IP leg
sip-profiles 1	Apply sip profile at global level

Please note that there is a difference between 12.4T and 15.4(3)M2 trains regarding two commands “header-passing” and “error-passthru”, which should be taken into account while making an update between the two IOS versions. With 12.4T they should be invoked together as "header-passing error-passthru" while in 15.4(3)M2 they should be invoked as 2 separate commands: "header-passing" and "error-passthru"

12.1.6.1 Media Passing through CUBE (media flow-through vs. media flow-around)

Default CUBE configuration enables CUBE to work in flow-through mode. In order to enable flow-around mode, please perform the following actions:

```
voice service voip
   media flow-around
```

12.1.6.2 Codecs

BT/BTIP requires currently monocodec configuration. That means, that only a single codec should be offered by CUBE. This is configured using codec class which is then applied to specific dial-peer.

For customers using **G.711 alaw** codec:

```
voice class codec 1
  codec preference 1 g711alaw
```

For customers using **G.711 ulaw** codec:

```
voice class codec 1
  codec preference 1 g711ulaw
```

12.1.6.3 SIP user agent

SIP signaling parameters are configured in the sip user agent section.

```
sip-ua
   retry invite 1
   retry response 2
   retry bye 2
   retry cancel 2
   reason-header override
   connection-reuse
   g729-annexb override
   timers options 1000
```

Explanation

Command	Description
retry …	Specifies number of retries for different SIP message types
reason-header override	Enable cause code passing from one SIP leg to another
connection-reuse	Always use the same port for both source and destination (UDP 5060)
g729-annexb override	Required for interoperability with BT/BTIP infrastructure,

	when G.729 codec is used

12.2 Integrating Sagem XMedius Fax Server Enterprise 8.0 with CUCM

In this section, we will present the steps necessary to integrate Sagem XMedius fax server with Cisco Unified Communications Manager (CUCM).

The XMediusFAX Enterprise edition is field proven to manage large fax volumes and deliver high levels of security, advanced integration, and monitoring & reporting capabilities. It is targeted for small and large enterprises and contains a number of key features.

12.2.1 Highlights for Sagem XMediusFax Server Enterprise 8.0.0.300:

- XMediusFAX is Sagemcom's innovative and patented IP fax server solution supporting the robust and standardized T.38 Fax over IP (FoIP) protocol.
- Direct SIP trunking with BTIP
- Simplified application integration through standardized technologies (i.e. XML, Python, Web Services API)
- Business critical system monitoring through application SNMP traps and PerfMon counters
- SQL database scalable to millions of inbound / outbound faxes with easy archiving
- Enhanced LDAP directory integration (i.e., Active Directory, Lotus Domino) with LDAPS support
- Intelligent fax boards and T.38 support
- Virtual machine support using VMware, Microsoft Hypervisor and Citrix
- Supported Document Formats: Adobe PDF, HTML, JPG, GIF, RTF, Microsoft Word, PowerPoint, Excel, Any Windows application that support "Print-To".
- Monitor all faxes sent, received, or in process, as well as server status
- Live graphical fax port usage monitor and integrated network packet capturing utility
- Email notification of service status events to administrator via SMTP
- Administrative audit logging and application services status changes logged in Windows Event Log
- System queue monitoring and alerts through SNMP and Performance Monitor (PerfMon)
- Integrated system reporting with a comprehensive set of 20+ built-in reports
- SSL authentication and encryption between all server modules and clients
- HTTPS for secured Web Client communications
- Built-in Windows Authentication support
- Support for LDAP over SSL (LDAPS)
- Enforce usage of billing codes
- Restricted destination fax number tables

- Per user/profile security settings (Allow to fax, require password, modify sender information, enforce cover page)

12.2.2 Supported fax features with BTIP Service

Please refer to the roadmap, the restriction portal and the INA synopsis portal for more information. List of supported features by XMediusFax Server Enterprise:

- Fax calls using G.711 a-law, G.711 u-law OR G.729 codec can only be proposed in case of specific offers (monocodec configuration – only one codec can be used in WAN for each customer)
- Send fax using XMediusFax SendFax desktop application
- Send fax using XMediusFax Web Panel application
- Incomming fax traffic
  - From standard G3/SG3 Fax machines
- Outgoing fax traffic
  - To standard G3/SG3 Fax machines.
- Sagem XmediusFax server can send G3 or SG3. This is global setting declared in license file and cannot be change without obtaining new license file.

12.3 Sagem XMediusFax Server components configuration

	Creating a Profile
Step 1	Immediately after installation, the **Basic** and **No Faxing Rights** profiles are available, to which you can associate users. The **Basic profile** allows the user to fax at a normal fax priority, with three retries if a connection cannot be immediately established The **No Faxing Rights profile** does not allow the transmission of faxes. You might also create new profiles and assign them to meet the specific fax needs of each user. It is also possible to create different profiles for each department, thereby tailoring fax settings to departmental requirements rather than user requirements.
	In the MMC Snap-in, select the **Profiles** node of your site, and click on the **Add** button.The **Profile Properties** dialog appears.

<table>
<tr><td></td><td>
<table>
<tr><th>Parameter Name</th><th>Parameter Value</th></tr>
<tr><td>❶ Enter the name of the profile In the Profile Name field.

❷ Select the Phone Books tab. If you want to assign phone books to the profile:
- In the Phone Books section, click Add. The Phone Book Properties dialog appears.
- Select a phone book in the Phone Book dropdown list.

Note: A phone book must have been previously created. To create and populate a phone book refer to the Administration Guide – Web documentation.

❸ Select the Billing Codes tab to Associating a Profile and a Billing Group - Once billing groups have been created, administrators can associate a billing group with a profile. The billing group can contain any number of billing codes and sub-billing codes which users can apply when faxing.

❹ Click the Fax Options tab to set the fax priority and how it affects the order in which the faxes are sent. This is however compounded by the number of retry attempts to send a fax.

❺ Select the Security tab to apply security settings.

❻ Select the Notification tab to set Notifications. By default, incoming fax notifications are sent to the destinations in the Incoming Routing Table, or to the default destination specified in its properties. Outbound fax notifications are sent to the sender's e-mail address.</td>
<td>❶ Sagem XMF Warsaw

❷ for example: 3580000

❸ Default values are used

❹ Default values are used

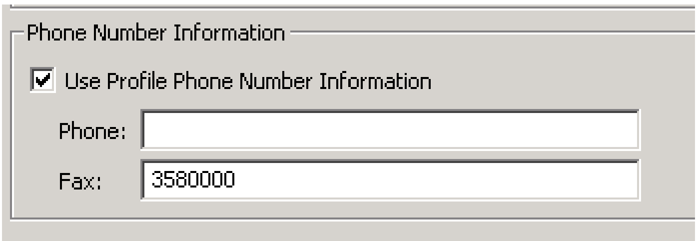
❺ Default values are used

❻ Default values are used</td></tr>
</table>
</td></tr>
<tr><td>Step 2</td><td>Sagem XMediusFax number presentation on SIP trunk
Configuration of number presentation on SIP trunk from XMF to CUCM. Number presentation – this number will be included in SIP</td></tr>
</table>

INVITE message send by Sagem server, for example:
SIP INVITE SDP() → *SIP From: sip:**3580000**@XMF_IP:5060*

Sites > Site_name > Configuration > Profiles > Profile properties > Profile tab > **Phone Number Information** section

Parameter Name	Parameter Value
❶ Phone Number Information section > Select **Profile Phone Number Information** checkbox	❶ checkbox must be enabled
❷ In **Fax** field provide phone number "extension" compliant with XMF dialplan	❷ for example: 3580000
❸ **Phone** field can be empty, not required to provide phone number	❸ empty value



Picture 2: Phone Number Information configuration in Profile

Step 3

Creating an Internal User Account

In the administration interface, select the **Internal User** node of your site and click on the **Add** button. The **User Properties** dialog appears.

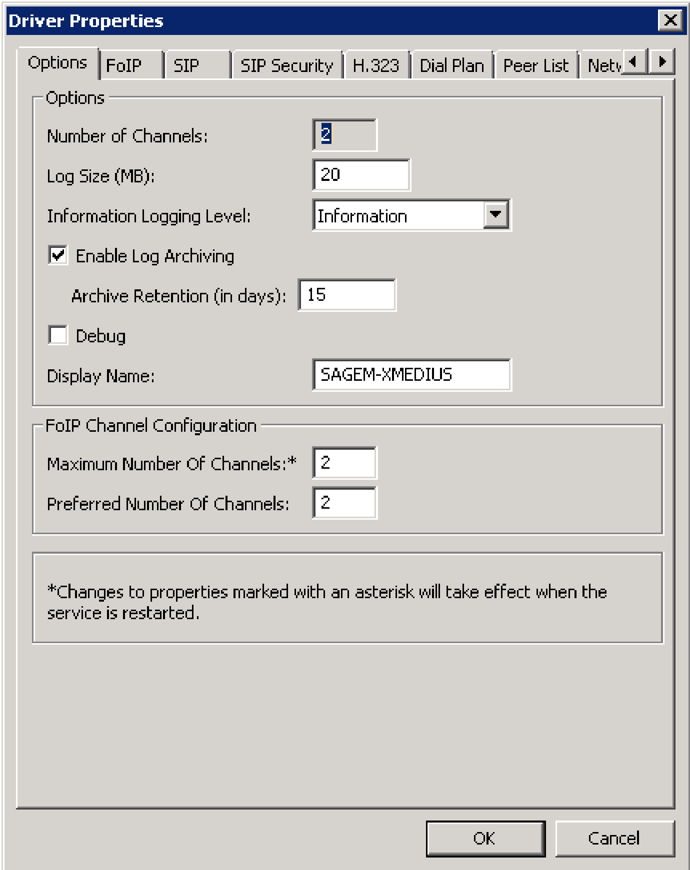
Parameter Name	Parameter Value
❶ Enter the **SMTP address** of the user; this is a mandatory entry.	❶ 3580001@orange-multimedia.fr
❷ Use **Profile Name** to associate the user to a specific profile. **Note:** A profile is mandatory. If no profile exists, you can choose Basic or No Faxing Rights. If you want to create a new profile, refer to **Step 1**. **Tips:** If the SMTP user has a corresponding Windows Domain account, use **AD account** to indicate that account in the format **domain\username**.	❷ Profile Name: **Basic**
❸ Navigate to **Personal Information** tab in User Properties windows. Provide **Phone Number Information** details (Phone number and Fax number) for	❸ Personal Information example: Phone: **3580001** Fax: **3580001**

	new user. Must be compliant with XMF dial plan.

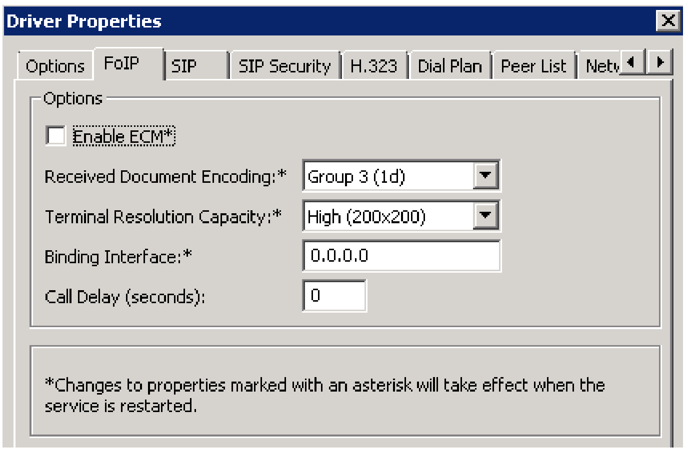
	T.38 Driver Properties Configuration (Options, T.38, SIP) In the administration interface, you just need to access the properties of the Driver node of your host to configure general SIP properties and to configure SIP specific properties for listed gateways and associate number patterns to specific gateway. **Warning:** Parameters locations on Driver Properties tabs can be different. It depends on T.38 driver release installed on the server.
Step 4	**System Configuration** > **Hosts** > **XMF_Host_name** > **Driver** container > Right Mouse Button click on **Driver** container and select **Properties**. In the **Driver properties** dialog, select the **Options** tab.

Parameter Name	Parameter Value
❶ On **Options** tab enable **Enable Log Archiving** property**.** Enables automatic log archiving for future support use.	❶ Checkbox **Enable Log Archiving** must be enabled. Set **Archive Retention** (in days) to value: **15.**
❷ On **Options** tab **Debug** checkbox should be disabled.	❷ Disabled
❸ On **Options** tab the T.38 Channel Configuration Section configuration.	❸ When you acquire a new license, you need to update here the number of channels allowed according to this new license
❹ On **FoIP** tab configure ECM (error correction mode).	❹ ECM may be enabled (Enabled ECM checkbox) or disabled. It depends on customer requirements. **If Enabled:** • **Received Document Encoding** set to **Group 3 (1d)** • **Terminal Resolution Capacity** set to **High (200x200)**
❺ In the **Driver properties** dialog, select the **SIP** tab. Provide port number under	❺ The general SIP properties are the following • Local SIP UDP Port - **5060**

which SIP messages are received for UDP, TCP and TLS.	• Local SIP TCP Port - **5060** • Local SIP TLS Port – **5061** • Print SIP Messages – **Disabled** • Wait For DTMF Code Input - **Disabled**



Picture 5: Example of Driver Configuration (Options tab)



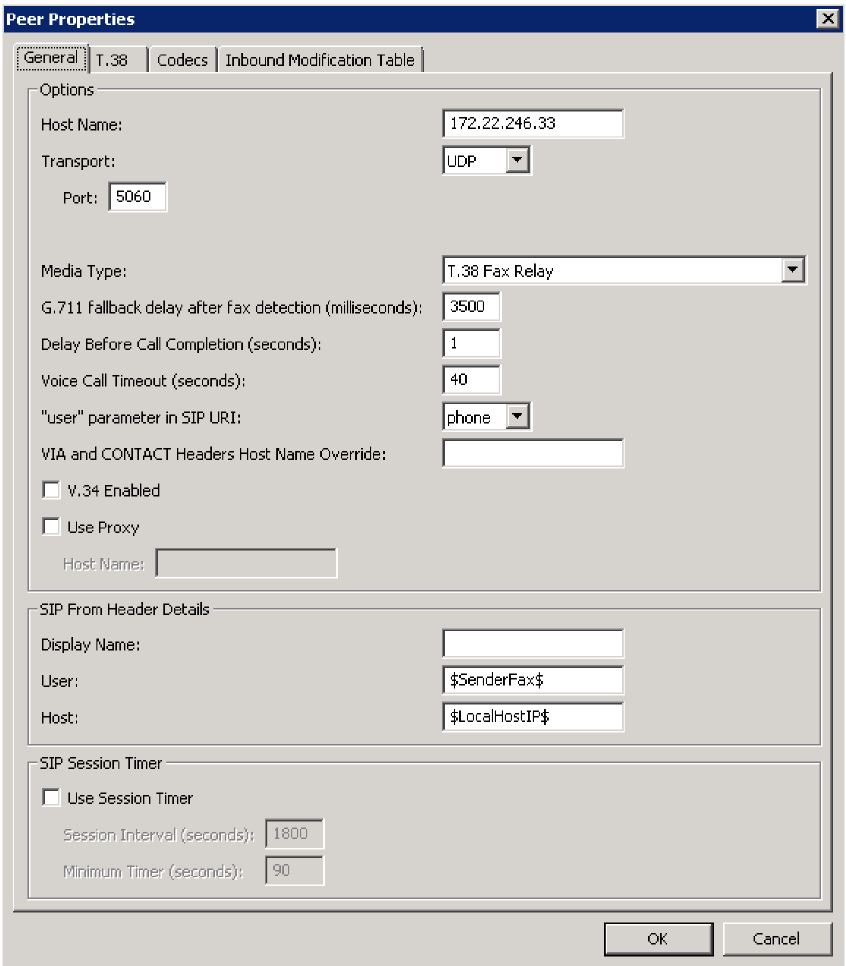
Picture 6: Example of Driver Configuration (FoIP tab) with Disabled ECM

	Note: If XmediusFAX is installed in high availability mode driver settings must be configured on all nodes visible in hosts list.

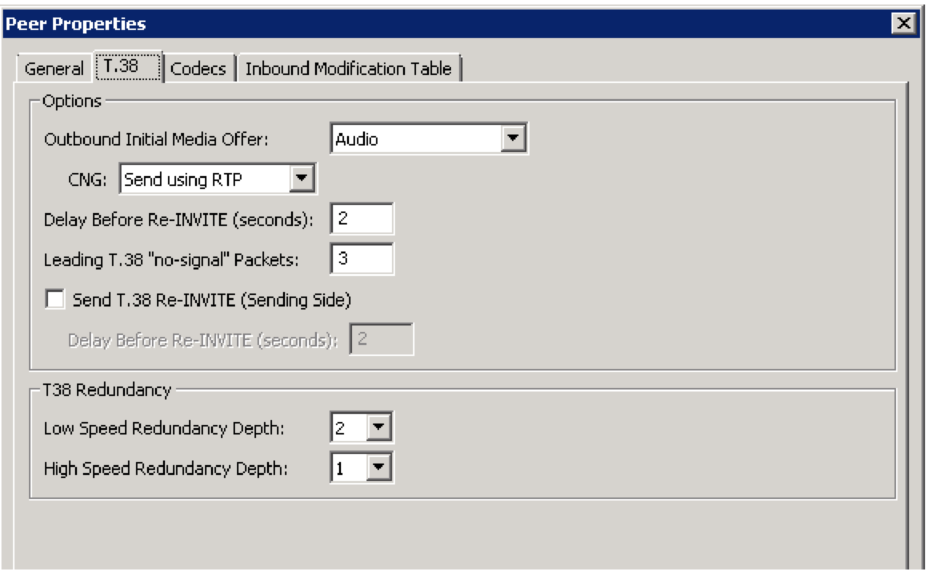
	T.38 Driver Properties Configuration (Managing a Dial Plan and Peer List) By default, XMediusFAX assumes that all faxes are to be sent through a single gateway. The list SIP gateways (in our case it will be CUCM), called the Peer List, therefore displays the single gateway established when XMediusFAX was installed. The corresponding dial plan indicates that all numbers will use the only gateway available. By using a Peer List, you can manage separately the SIP or H.323 properties to use for each known gateway (or proxy) that communicate with the fax server.
Step 6	**System Configuration** > **Hosts** > **XMF_Host_name** > **Driver** container > Right Mouse Button click on **Driver** container and select **Properties**.
	In the **Driver properties** dialog, select the **Peer List** tab.

Parameter Name	Parameter Value
❶ Click **Add SIP Peer** button. Adds a new SIP Peer and allows to configure its properties	❶ Checkbox **Enable Log Archiving** must be enabled. Set **Archive Retention** (in days) to value: **15**.
❷ On **General** tab of Peer Properties window provide **Host Name** - The host name of the gateway (or proxy) to be added as a Peer.	❷ IP address of CUCM, for example: **6.5.6.1**.
❸ On **General** tab of Peer Properties window provide the transport type (UDP, TCP or TLS) to be used by this Peer.	❸ Transport: **UDP**
❹ On **General** tab of Peer Properties window provide the port number of this Peer.	❹ 5060
❺ On **General** tab of **Delay Before Call Completion, Voice Call Timeout and SIP From Header Details.**	❺ Delay Before Call Completion – **1 second** Voice Call Timeout – **40 seconds** Display name – **empty** User - **$SenderFax$**

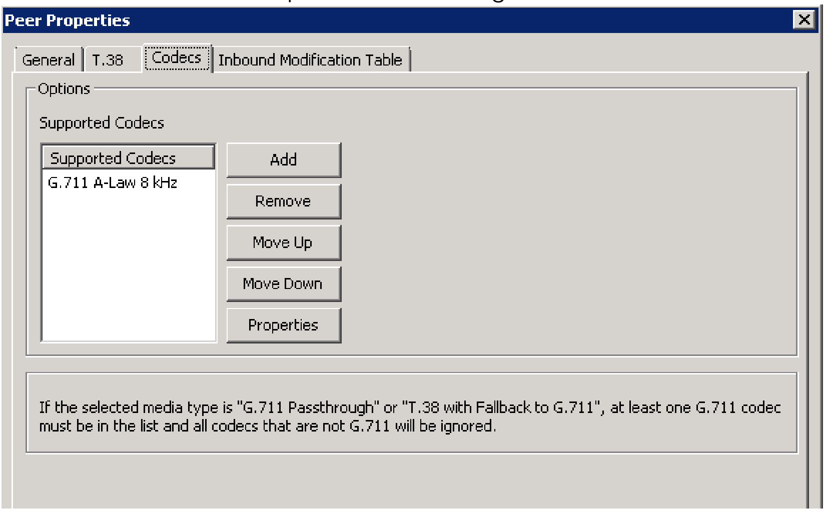
❻ On **T.38** tab of Peer Properties window configure **Outbound Initial Media Offer** and **CNG** options.	Host - **$LocalHostIP$** ❻ Outbound Initial Media Offer -**Audio** CNG - **Send CNG using RPT**
❼ On **T.38** tab of Peer Properties window configure **Delay before Re-INVITE.**	❼ Delay before Re-INVITE - **2 seconds**
❽ On **T.38** tab of Peer Properties window configure properties of the **T38 redundancy** section.	❽ LS redundancy (possible range 0-2) – **2** HS redundancy (possible range 0-2) – **1**
❾ On **Codecs** tab click **Add** button to choose codec from **Available Codecs** list.	❾It depends on codec requirements, three supported possibilities by Orange Infrastructure: - **G.711 A-Law 8 kHz** - **G.711 u-law 8 kHz** - or G.729 8kHz



Picture 7: Example of Driver Configuration – new Peer SIP From Headers configuration



Picture 8: Example of Driver Configuration - new Peer

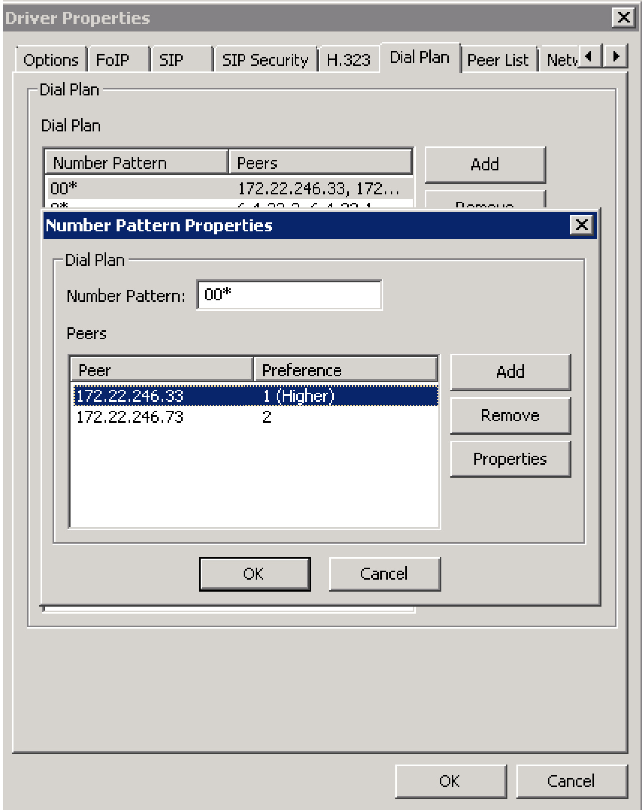


Picture 9: Example of Driver Configuration – new Peer Codec

In the **Driver properties** dialog, select the **Dial Plan** tab.

Parameter Name	Parameter Value
❶ Click **Add** button. Provide **number pattern** you wish to associate with the list of Peers below.	❶ * (asterisk) **Note:** You must specify the entire fax number anticipated. Wildcards can be entered: - The asterisk (*) specifies any number of digits - The question mark (?) specifies a single digit.

❷ Select a Peer to Add to the List Associated with a Number Pattern. Click Add button to select configured Peer (Orange SBC).	❷ Peer: **6.5.6.1** **Preference: 1 (Higher)**
❸ On **General** tab of Peer Properties window provide the transport type (UDP, TCP or TLS) to be used by this Peer.	❸ Transport: **UDP**



Picture 10: Example of Driver Configuration – Dial Plan configuration

Note: If XmediusFAX is installed in high availability mode driver settings must be configured on all nodes visible in hosts list.

	Incoming routing table (System Configuration)
Step 7	**XMediusFax > System Configuration > Hosts > Incoming Routing Table** In the MMC Snap-in, select the **Incoming Routing Table** node and then click **Add.** The **Routing Table Entry Properties** dialog appears

Parameter Name	Parameter Value
❶ Enter a valid DNIS/DID number in the Lower Bound field.	❶ 3580000

	❷ Enter a valid DNIS/DID number in the Upper Bound field.	❷ 3580099 Note: The Lower Bound and Upper Bound values must have the same amount of digits and the Upper Bound value must be higher than the Lower Bound value.
	❸ Select the site to which you want to associate these values, from the list in the Site field.	❸ Site : Sagem
	❹ Enter the site Call Station ID in the CSID field.	❹ CSID : sagem

12.3.1 CUCM Configuration

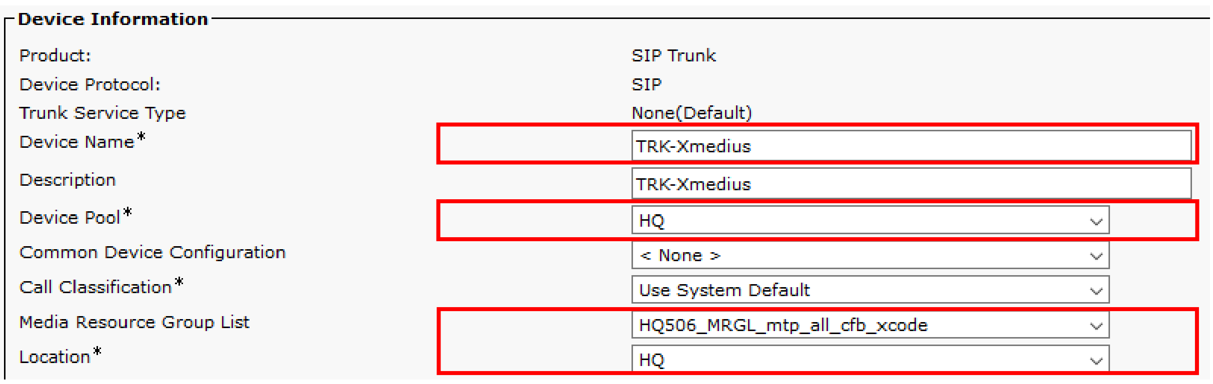
This section describes the steps necessary to take on CUCM in order to integrate it with Sagem Xmedius Fax server.

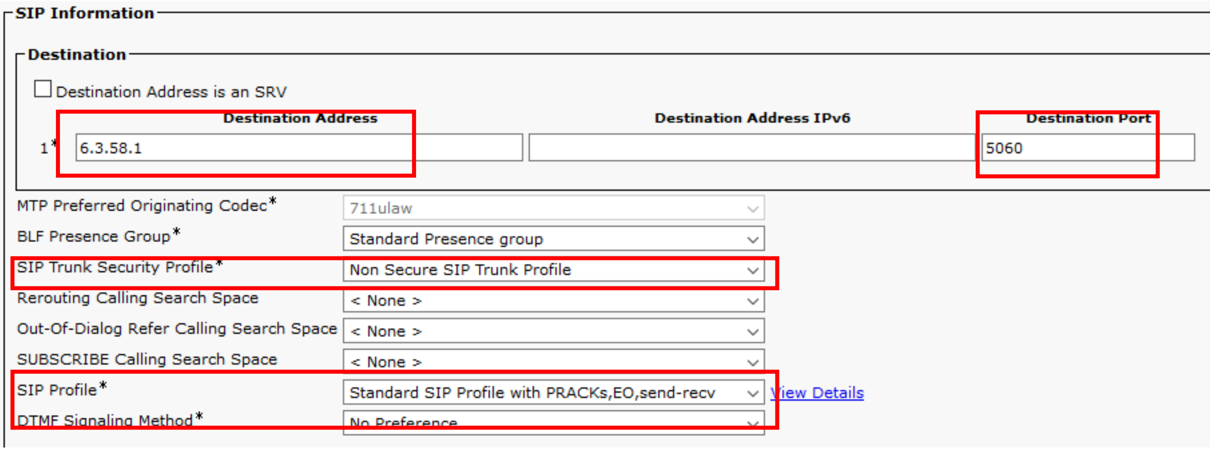
12.3.1.1 SIP Trunk Configuration

Go to Device -> Trunk and click Add New. On next page, select following options:

- Trunk Type: SIP Trunk
- Device Protocol: SIP
- Trunk Service Type: None (Default)

Click Next. In next window, configure following options:



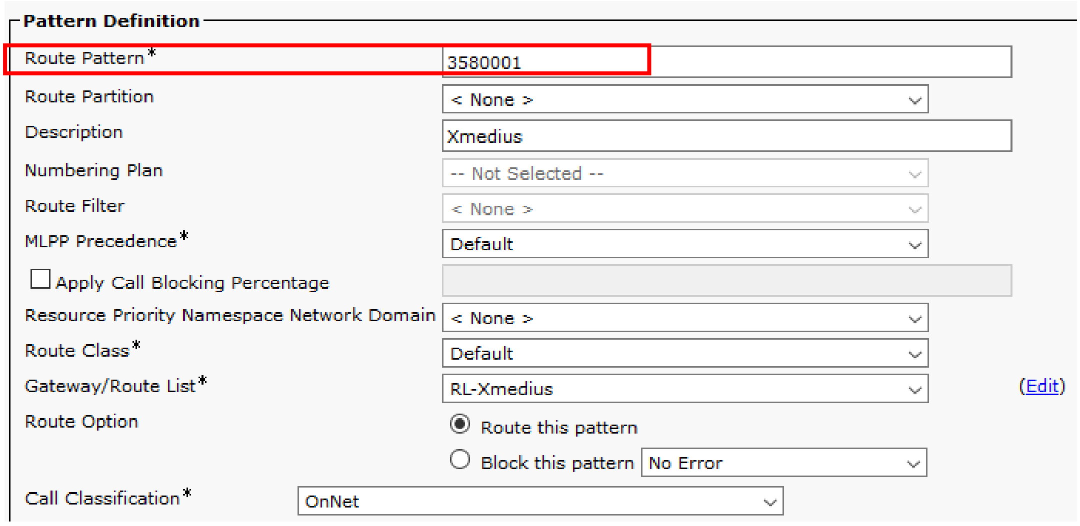


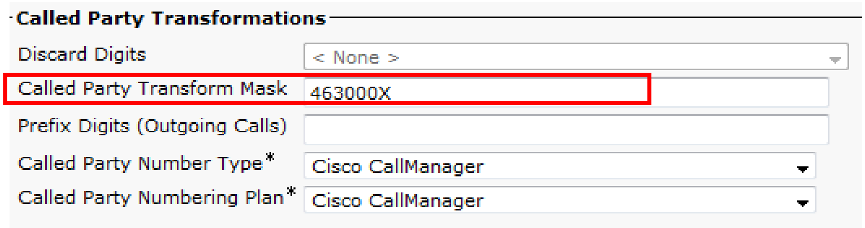
Setting	Value	Description
Device Name	TRK-Xmedius	Name of SIP Trunk
Device Pool	HQ	Device Pool, to which this SIP Trunk belongs
Media Resource Group List	MRGL_MTP_XCODE	Select MRGL which has MTPs, transcoders and other standard media resources.
Destination Address	IP Address of Sagem Xmedius	Specify the IP address of Sagem Xmedius Fax server
Destination Port	5060	Specify the port, which will be used for communication, 5060 is default one.
SIP Trunk Security Profile	Non-Secure SIP Trunk Profile	Standard, built-in SIP Trunk Security Profile.
SIP Profile	Standard SIP Profile with PRACKs, EO, send-recv	Standard SIP Profile.
DTMF Signalling Method	No Preference	Chooses any compliant method of DTMF signals transport.

Select Save - this finishes configuration of SIP Trunk.

12.3.1.2 Route Pattern Configuration

In order to have calls routed to Sagem Xmedius, we need to configure the dial-plan element which will allow this. Go to Call Routing -> Route/Hunt > Route Pattern. Click Add New button and configure following options:





Setting	Value	Description
Route Pattern	Depends on deployment Here: 3580001	Dialed number that will be directed to Sagem Xmedius fax server.
Called Party Transform Mask	Depends on deployment Here: 463000X	Called number to which originally dialed number will be transformed to. Can be left blank if no change required.

Confirmation tests

12.4 Validation overview

The complete FAX gateway/endpoint validation consists of

1. Functional tests – mix of tests using G3 and Super G3 machines in both directions. Engineering confirms overall page transmission quality (visual comparison) and technical aspects like T38 profile, transmission speed, T30 negotiation and fallback to G3.

2. Statistical tests – stress tests of device. FaxLab application connected to ChannelTrap simulators repeats fax calls many times to confirm device stability in longer period of time.

12.5 Validation

12.5.1 Functional

It is a list of incoming and outgoing FAX calls going through Business Talk infrastructure. Following tests should be done using non empty page (full text or simple image).

Test Distribution		
Direction	**Gateway**	**PSTN Fax**
Incoming	G3	G3
Outgoing	G3	G3
Incoming	SG3	G3
Outgoing	SG3	G3
Incoming	G3	SG3
Outgoing	G3	SG3
Incoming	SG3	SG3
Outgoing	SG3	SG3

12.5.2 Statistical

Statistical tests have been done to confirm live implementation stability. Statistical session as described in following table:

Type of calls		Number of pages
Fax type	**Direction**	**10p**
G3	Incoming	100x
	Outgoing	100x
SG3	Incoming	100x
	Outgoing	100x

ANNEX A: Provisioning Oracle ESBC

1.1 Global configuration

1.1.1 Media configuration

1.1.1.1 Media Manager Configuration

Define the max-bandwidth available for signaling and specify the name of the 'home-realm'.

The max-bandwidth is a self-protection parameter set to 1767740 for AP4500 and 2351094 for AP4600 (as per Oracle best practices) in order to prevent the SBC from being overwhelmed by a too big volume of incoming sip traffic. The untrusted signaling bandwidth is set to a minimal value as such traffic is not expected. A specific bandwidth is reserved for fragmented packets.

Note: a reboot is necessary after the modification of the media-manager parameters.

Element	Configuration
Media Manager Configuration	CSBC# conf t CSBC(configure)# media-manager CSBC(media-manager)# media-manager CSBC(media-manager-config)# select CSBC(media-manager-config)# max-signaling-bandwidth 1767740 for AP4500 or 2351094 for AP4600 CSBC(media-manager-config)# anonymous-sdp enabled CSBC(media-manager-config)# max-untrusted-signaling 1 CSBC(media-manager-config)# min-untrusted-signaling 1 CSBC(media-manager-config)# fragment-msg-bandwidth 90000 for AP4500 only CSBC(media-manager-config)# options hairpin-released-flows CSBC(media-manager-config)# options +dont-terminate-assoc-legs CSBC(media-manager-config)# done media-manager state enabled latching enabled flow-time-limit 86400 initial-guard-timer 300 subsq-guard-timer 300 tcp-flow-time-limit 86400 tcp-initial-guard-timer 300 tcp-subsq-guard-timer 300 tcp-number-of-ports-per-flow 2 hnt-rtcp disabled algd-log-level NOTICE mbcd-log-level NOTICE options dont-terminate-assoc-legs hairpin-released-flows red-flow-port 1985 red-mgcp-port 1986 red-max-trans 10000 red-sync-start-time 5000 red-sync-comp-time 1000 media-policing enabled max-signaling-bandwidth 1767740 max-untrusted-signaling 1

	min-untrusted-signaling 1 app-signaling-bandwidth 0 tolerance-window 30 trap-on-demote-to-deny enabled trap-on-demote-to-untrusted disabled syslog-on-demote-to-deny disabled syslog-on-demote-to-untrusted disabled rtcp-rate-limit 0 anonymous-sdp enabled arp-msg-bandwidth 32000 fragment-msg-bandwidth 90000 rfc2833-timestamp disabled default-2833-duration 100 rfc2833-end-pkts-only-for-non-sig enabled translate-non-rfc2833-event disabled media-supervision-traps disabled dnsalg-server-failover disabled syslog-on-call-reject disabled last-modified-by admin@10.10.0.5 last-modified-date 2015-10-01 15:18:05 CSBC(media-manager-config)# exit CSBC(media-manager)# exit CSBC(configure)#

1.1.2 Codec Policy

One codec policy is created to filter out audio codecs (and their parameters) different from G.722, G.711 and G.729, to allow telephone-events and T.38 and to disable the video media. This policy is called by the access realm. A second codec policy is created to add G.711µlaw support and is called by the Core realm.

In addition a customer specific codec policy can be added.

Element	Configuration

Codec Policy	CSBC# conf t
	CSBC(configure)# media-manager
	CSBC(media-manager)# codec-policy
	CSBC(codec-policy)# name codecfiltering
	CSBC(codec-policy)# allow-codecs (PCMA G722 G729 telephone-event t.38 video:no)
	CSBC(codec-policy)# done
	codec-policy
	name codecfiltering
	allow-codecs PCMA G722 G729 telephone-event t.38 video:no
	order-codecs
	last-modified-by admin@10.10.0.5
	last-modified-date 2013-04-16 16:29:28
	CSBC(codec-policy)# name codecfilteringCore
	CSBC(codec-policy)# allow-codecs (PCMA PCMU G722 G729 telephone-event t.38 video:no)
	CSBC(codec-policy)# done
	codec-policy
	name codecfilteringCore
	allow-codecs PCMA PCMU G722 G729 telephone-event t.38 video:no
	order-codecs

	last-modified-by admin@10.10.0.5 last-modified-date 2014-01-23 15:44:50 CSBC(codec-policy)# allow-codecs (PCMU telephone-event t.38 video:no) CSBC(codec-policy)# done codec-policy name codecPCMU allow-codecs PCMU telephone-event t.38 video:no order-codecs last-modified-by admin@10.10.0.5 last-modified-date 2014-01-23 15:44:50 CSBC(codec-policy)# exit CSBC(media-manager)# exit CSBC(configure)#

1.1.2.1 Media Security Policy

This is required for the Oracle ESBC Media security policy which not required encryption.

A media-security-policy is configured and is called in the Core realm. It defines that media is not encrypted in the Core. This configuration is required when at least one access customer realm is configured to support SRTP.

Element	Configuration
Codec Policy	CSBC# conf t CSBC(configure)# security media-security media-sec-policy CSBC(media-sec-policy)# name nocrypto CSBC(media-sec-policy)# inbound CSBC(media-sec-inbound)# mode rtp CSBC(media-sec-inbound)# done inbound profile mode rtp protocol none CSBC(media-sec-inbound)# exit CSBC(media-sec-policy)# outbound CSBC(media-sec-outbound)# mode rtp CSBC(media-sec-outbound)# done outbound profile mode rtp protocol none CSBC(media-sec-outbound)# exit CSBC(media-sec-policy)# done media-sec-policy name nocrypto pass-through disabled options inbound profile mode rtp protocol none outbound profile mode rtp

	protocol none last-modified-by admin@10.10.0.5 last-modified-date 2015-08-26 12:16:59

1.1.3 Global Sip Configuration

1.1.3.1 User-Agent

Within OBS VISIT SIP certification program context, User agent header must have following format:

User-Agent: ORACLE <SBC Model>/v.8.2.0 \\ Cisco-CUCM12.0

1.1.3.2 Sip-config

Set these general parameters to allow the correct functioning of the ESBC.

The trans-expire and initial-inv-trans-expire timers are set to 5 seconds (this replaces the default 32 seconds) which entails that a SIP request will be transmitted up to 4 times before it expires.

Three options have been added:

- max-udp-length=0 option forces ESBC to send fragmented IP packets when UDP datagram size exceeds 1500B
- set-inv-exp-at-100-resp stops the trans-expire timer after receiving the 100Trying provisional response
- sag-target-uri=ip makes the IP address instead of its hostname be used in request-lines of messages sent by the ESBC to a session-agent.

Element	Configuration
Sip-config	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# sip-config CSBC(sip-config)# select CSBC(sip-config)# home-realm-id Core CSBC(sip-config)# nat-mode None CSBC(sip-config)# registrar-domain * CSBC(sip-config)# registrar-host * CSBC(sip-config)# registrar-port 5060 CSBC(sip-config)# trans-expire 5 CSBC(sip-config)# initial-inv-trans-expire 5 CSBC(sip-config)# invite-expire 200 CSBC(sip-config)# options +max-udp-length=0 CSBC(sip-config)# options +sag-target-uri=ip CSBC(sip-config)# options +set-inv-exp-at-100-resp CSBC(sip-config)# done sip-config state enabled operation-mode dialog dialog-transparency enabled home-realm-id Core egress-realm-id auto-realm-id nat-mode None registrar-domain * registrar-host * registrar-port 5060 register-service-route always init-timer 500 max-timer 4000 trans-expire 5 initial-inv-trans-expire 5

	invite-expire 200 session-max-life-limit 0 inactive-dynamic-conn 32 enforcement-profile pac-method pac-interval 10 pac-strategy PropDist pac-load-weight 1 pac-session-weight 1 pac-route-weight 1 pac-callid-lifetime 600 pac-user-lifetime 3600 red-sip-port 1988 red-max-trans 10000 red-sync-start-time 5000 red-sync-comp-time 1000 options max-udp-lenght=0 sag-target-uri=ip set-inv-exp-at-100-resp add-reason-header disabled sip-message-len 4096 enum-sag-match disabled extra-method-stats enabled extra-enum-stats disabled mps-volte disabled rph-feature disabled nsep-user-sessions-rate 0 nsep-sa-sessions-rate 0 registration-cache-limit 0 register-use-to-for-lp disabled refer-src-routing enabled add-ucid-header disabled proxy-sub-events allow-pani-for-trusted-only disabled atcf-stn-sr atcf-psi-dn atcf-route-to-sccas disabled eatf-stn-sr pass-gruu-contact disabled sag-lookup-on-redirect disabled set-disconnect-time-on-bye disabled msrp-delayed-bye-timer 15 transcoding-realm transcoding-agents create-dynamic-sa disabled node-functionality P-CSCF match-sip-instance disabled sa-routes-stats disabled sa-routes-traps disabled rx-sip-reason-mapping disabled add-ue-location-in-pani disabled hold-emergency-calls-for-loc-info 0 retry-after-upon-offline 0 reg-reject-response-upon-offline 503 hold-invite-calls-for-loc-info 0 cache-loc-info-expire 32 msg-hold-for-loc-info 0 npli-upon-register disabled last-modified-by admin@10.253.51.48

	last-modified-date 2019-03-26 13:46:14 CSBC(sip-config)# exit CSBC(session-router)# exit CSBC(configure)#

1.1.3.3 Header Whitelists

Header whitelists remove all SIP headers that are not defined in the list, from the SIP messages. This feature enables to clean up the SIP messages by deleting vendor specific headers or other headers that are not useful for the BTIP/BT service, hence facilitating the interoperability between IPBXs or between BTIP/BT and other services.

Three whitelists are defined:

- headersWLaccess: filters out headers from the messages coming from the access South side to the ESBC.
- headersWLcore: filters out headers from the messages coming from the Core North side to the ESBC.

Note that the whitelists are applied after the HMR of the ingress direction.

When modifying the allow-any parameter, you can add and delete single entries from the list using plus (+) and minus (-) signs without having to overwrite the whole list.

Element	Configuration

Sip Headers IPBX Access South side Whitelists	CSBC# conf t
	CSBC(configure)# session-router
	CSBC(session-router)# allowed-elements-profile
	CSBC(allowed-elements-profile)# name headersWLAccess
	CSBC(allowed-elements-profile)# allow-any (Accept Allow Allow-Events Call-ID Contact Content-Disposition Content-Length Content-Type CSeq Diversion Event Expires From History-Info Max-Forwards Privacy RAck Reason Record-Route Request-uri Require Route RSeq Subscription-State Supported To Via User-Agent Server P-Early-Media P-identifier Unsupported User-To-User Warning MIME-version Remote-Party-ID Timestamp)
	CSBC(allowed-elements-profile)# allow-any +P-Initial-Asserted-Id
	CSBC(allowed-elements-profile)# allow-any +P-Options
	CSBC(allowed-elements-profile)# allow-any +P-Initial-From-User
	CSBC(allowed-elements-profile)# rule-sets
	CSBC(allowed-rule-sets)# name ruleCSeq
	CSBC(allowed-rule-sets)# unmatched-action delete
	CSBC(allowed-rule-sets)# done
	rule-sets
	name ruleCSeq
	unmatched-action delete
	msg-type any
	methods
	logging disabled
	CSBC(allowed-rule-sets)# exit
	CSBC(allowed-elements-profile)# done
	allowed-elements-profile
	name headersWLAccess
	description
	rule-sets
	name ruleCSeq
	unmatched-action delete
	msg-type any
	methods

	logging disabled allow-any Accept Allow Allow-Events Call-ID Contact Content-Disposition Content-Length Content-Type CSeq Diversion Event Expires From History-Info Max-Forwards MIME-version P-Early-Media P-identifier P-Initial-Asserted-Id P-Initial-From-User P-Options Privacy RAck Reason Record-Route Remote-Party-ID Request-uri Require Route RSeq Server Subscription-State Supported Timestamp To Unsupported User-Agent User-To-User Via Warning last-modified-by admin@10.253.51.48 last-modified-date 2019-09-17 14:53:11

Sip Headers Core North BTIP/BT Whitelists	CSBC# conf t CSBC(allowed-elements-profile)# name headersWLCore CSBC(allowed-elements-profile)# CSBC(allowed-elements-profile)# allow-any (Accept Allow Allow-Events Call-ID Contact Content-Disposition Content-Length Content-Type CSeq Diversion Event Expires From History-Info Max-Forwards P-Access-Network-Info P-Asserted-Identity Privacy RAck Reason Record-Route Request-uri Require Route RSeq Subscription-State Supported To Via P-Early-Media Unsupported User-To-User Warning MIME-version Remote-Party-ID Timestamp) CSBC(allowed-elements-profile)# rule-sets CSBC(allowed-rule-sets)# unmatched-action delete CSBC(allowed-rule-sets)# name ruleCSeq CSBC(allowed-rule-sets)# done allowed-elements-profile name headersWLCore description rule-sets name ruleCSeq unmatched-action delete msg-type any methods logging disabled allow-any Accept Allow Allow-Events Call-ID Contact Content-Disposition Content-Length Content-Type CSeq Diversion Event Expires From History-Info Max-Forwards MIME-version P-Access-Network-Info P-Asserted-Identity P-Early-Media Privacy RAck Reason Record-Route Remote-Party-ID Request-uri Require Route RSeq Subscription-State Supported Timestamp To Unsupported User-To-User Via

	Warning last-modified-by admin@10.253.51.48 last-modified-date 2019-09-17 14:57:50

1.1.3.4 SIP enforcement Profile

In order to reject unwanted methods with error response '405 Method Not Allowed' we configure an enforcement-profile to be applied on the receiving sip-interface. The enforcement profile includes the reference to the header whitelists.

Two profiles are defined:

- Filtermsg: defined in each access realm. It includes the whitelist headersWLAccess
- filterHeadersCore: defined in the Core realm. It includes the whitelist headersWLCore.

Element	Configuration
enforcement-profile for South IPBX Access side	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# enforcement-profile CSBC(enforcement-profile)# name filtermsg CSBC(enforcement-profile)# allowed-methods INVITE,PRACK,OPTIONS,UPDATE,,NOTIFY,INFO CSBC(enforcement-profile)# allowed-elements-profile headersWLAccess CSBC(enforcement-profile)# done enforcement-profile name filtermsg allowed-methods INVITE,PRACK,OPTIONS,UPDATE, NOTIFY,INFO sdp-address-check disabled allowed-elements-profile headersWLAccess add-certificate-info verify-certificate-info-register disabled certificate-ruri-check disabled last-modified-by admin@10.253.51.48 last-modified-date 2019-03-20 14:34:39
enforcement-profile for North BT/Btalk Core side	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# enforcement-profile CSBC(enforcement-profile)# name filterHeadersCore CSBC(enforcement-profile)# allowed-methods INVITE,PRACK,OPTIONS,UPDATE,NOTIFY,INFO CSBC(enforcement-profile)# allowed-elements-profile headersWLCore CSBC(enforcement-profile)# done enforcement-profile name filterHeadersCore allowed-methods INVITE,PRACK,OPTIONS,UPDATE,NOTIFY,INFO sdp-address-check disabled allowed-elements-profile headersWLCore add-certificate-info certificate-ruri-check disabled last-modified-by admin@10.10.0.5 last-modified-date 2013-07-01 14:23:51

1.1.3.5 SIP features

In order to accept requests requesting support of 100rel extension from and to any realm, and to reject requests requiring the support of timer and replaces extensions which are supported by OBS Btalk/BT offers, we need to create the following sip-feature:

Element	Configuration
Sip Features	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# sip-feature CSBC(sip-feature)# name 100rel CSBC(sip-feature)# require-mode-inbound pass CSBC(sip-feature)# require-mode-outbound pass CSBC(sip-feature)# done sip-feature name 100rel realm 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 last-modified-by admin@10.253.51.48 last-modified-date 2019-03-20 14:35:49 CSBC(sip-feature)# name timer CSBC(sip-feature)# support-mode-inbound strip CSBC(sip-feature)# require-mode-inbound reject CSBC(sip-feature)# proxy-require-mode-inbound reject CSBC(sip-feature)# support-mode-outbound strip CSBC(sip-feature)# require-mode-outbound reject CSBC(sip-feature)# proxy-require-mode-outbound reject CSBC(sip-feature)# done sip-feature name timer realm support-mode-inbound Strip require-mode-inbound Reject proxy-require-mode-inbound Reject support-mode-outbound Strip require-mode-outbound Reject proxy-require-mode-outbound Reject last-modified-by admin@10.253.51.48 last-modified-date 2019-03-20 14:36:23 CSBC(sip-feature)# name replaces CSBC(sip-feature)# support-mode-inbound strip CSBC(sip-feature)# require-mode-inbound reject CSBC(sip-feature)# proxy-require-mode-inbound reject CSBC(sip-feature)# support-mode-outbound strip CSBC(sip-feature)# require-mode-outbound reject CSBC(sip-feature)# proxy-require-mode-outbound reject CSBC(sip-feature)# done sip-feature name replaces

	realm support-mode-inbound Strip require-mode-inbound Reject proxy-require-mode-inbound Reject support-mode-outbound Strip require-mode-outbound Reject proxy-require-mode-outbound Reject last-modified-by admin@10.10.0.5 last-modified-date 2013-07-08 15:58:17

1.1.3.6 Response maps

Response maps enable to change the change the value of SIP error codes either generated by the ESBC or received by the ESBCs.

181 and 182 messages received by the ESBC are changed in 183 “Session Progress”. 403, 500 and 503 errors generated by the main BT/BTIP SIP termination (typically for a session-agent out of service or missing) are changed in 408 in order to trigger rerouting to the backup BT/BTIP SIP termination.

Element	Configuration
Core North BT Response maps	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# sip-response-map CSBC(response-map)# name BT CSBC(response-map)# entries CSBC(response-map-entry)# recv-code 181 CSBC(response-map-entry)# xmit-code 183 CSBC(response-map-entry)# reason "Session Progress" CSBC(response-map-entry)# done response-map name BT entries recv-code 181 xmit-code 183 reason Session Progress method register-response-expires CSBC(response-map-entry)# recv-code 182 CSBC(response-map-entry)# xmit-code 183 CSBC(response-map-entry)# reason "Session Progress" CSBC(response-map-entry)# done response-map-entry entries recv-code 182 xmit-code 183 reason Session Progress method register-response-expires CSBC(response-map-entry)# exit CSBC(response-map)# done response-map last-modified-by admin@10.253.51.48 last-modified-date 2019-09-18 15:01:39 name BT entries 181 -> 183 (Session Progress) 182 -> 183 (Session Progress)

Element	Configuration
Access South local Response maps	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# sip-response-map CSBC(response-map)# name localBT CSBC(response-map)# entries CSBC(response-map-entry)# recv-code 503 CSBC(response-map-entry)# xmit-code 408 CSBC(response-map-entry)# reason "Next-hop Unavailable" CSBC(response-map-entry)# done response-map-entry recv-code 503 xmit-code 408 reason Next-hop Unavailable method register-response-expires CSBC(response-map-entry)# recv-code 403 CSBC(response-map-entry)# xmit-code 408 CSBC(response-map-entry)# reason "Next-hop Unavailable" CSBC(response-map-entry)# done

1.2 Business Talk/ BTIP OBS Carrier North SIP configuration for Oracle ESBC configuration

1.2.1 Unsecured SIP Trunk through UDP

1.2.1.1 Core realm Configuration

A core realm (id 'Core') is created once to represent the OBS carrier part of the ESBC and provides media-ports sharing. This realm is associated with a SIP-interface that is common for all iPBX.

Note that SIP and media packets toward the PBX and the SSW will be marked according to the specified media-policy and class-profile.

A codec policy is defined to filter out video media, and unnecessary audio codecs and their parameters which are not supported by BT/BTIP offers.

For the AP4600 only, the parameter media-sec-policy is configured with the 'nocrypto' value.

Element	Configuration
Core Realm	CSBC# conf t CSBC(configure)# media-manager CSBC(media-manager)# realm-config CSBC(realm-config)# identifier Core CSBC(realm-config)# network-interfaces M00:<SBC_CORE_VLAN_ID> *ex: M00:20* CSBC(realm-config)# media-policy mark-mp CSBC(realm-config)# class-profile mark-cp CSBC(realm-config)# access-control-trust-level high CSBC(realm-config)# codec-policy codecfilteringCore CSBC(realm-config)# media-sec-policy nocrypto *For the AP4600 only* CSBC(realm-config)# done realm-config identifier Core description addr-prefix 0.0.0.0 network-interfaces M00:187 mm-in-realm disabled mm-in-network enabled mm-same-ip enabled mm-in-system enabled bw-cac-non-mm disabled msm-release disabled qos-enable disabled max-bandwidth 0 fallback-bandwidth 0 max-priority-bandwidth 0 max-latency 0 max-jitter 0 max-packet-loss 0 observ-window-size 0 parent-realm dns-realm media-policy mark-mp media-sec-policy nocrypto *For AP4600 only* rtcp-mux disabled ice-profile

```
dtls-srtp-profile
srtp-msm-passthrough           disabled
class-profile                  mark-cp
in-translationid
out-translationid
in-manipulationid
out-manipulationid
average-rate-limit             0
access-control-trust-level     high
invalid-signal-threshold       0
maximum-signal-threshold       0
untrusted-signal-threshold     0
nat-trust-threshold            0
max-endpoints-per-nat          0
nat-invalid-message-threshold  0
wait-time-for-invalid-register 0
deny-period                    30
session-max-life-limit         0
cac-failure-threshold          0
untrust-cac-failure-threshold  0
ext-policy-svr
diam-e2-address-realm
subscription-id-type           END_USER_NONE
symmetric-latching             disabled
pai-strip                      disabled
trunk-context                  -
device-id
early-media-allow
enforcement-profile
additional-prefixes
restricted-latching            none
restriction-mask               32
user-cac-mode                  none
user-cac-bandwidth             0
user-cac-sessions              0
icmp-detect-multiplier         0
icmp-advertisement-interval    0
icmp-target-ip
monthly-minutes                0
options
spl-options
accounting-enable              enabled
net-management-control         disabled
delay-media-update             disabled
refer-call-transfer            disabled
hold-refer-reinvite            disabled
refer-notify-provisional       none
dyn-refer-term                 disabled
codec-policy
codec-manip-in-realm           disabled
codec-manip-in-network         enabled
rtcp-policy
constraint-name
session-recording-server
session-recording-required     disabled
manipulation-string
manipulation-pattern
stun-enable                    disabled
stun-server-ip                 0.0.0.0
```

	stun-server-port 3478 stun-changed-ip 0.0.0.0 stun-changed-port 3479 sip-profile flow-time-limit -1 initial-guard-timer -1 subsq-guard-timer -1 tcp-flow-time-limit -1 tcp-initial-guard-timer -1 tcp-subsq-guard-timer -1 sip-isup-profile match-media-profiles qos-constraint block-rtcp disabled hide-egress-media-update disabled tcp-media-profile monitoring-filters node-functionality default-location-string alt-family-realm pref-addr-type none sm-icsi-match-for-invite sm-icsi-match-for-message ringback-trigger none ringback-file last-modified-by admin@10.253.51.48 last-modified-date 2019-09-12 13:10:32

1.2.1.2 Core realm sip-interface

A sip-interface must be associated to the Core realm previously defined with the port 5060. Only one SIP interface is used on Core to represent all Cisco IPBX.

The parameter '*allow-anonymous agents-only*' enables only a provisioned session-agent to send requests to the ESBC: messages received from unknown sources will be rejected with '403 Forbidden'.

Headers *P-charging-vector* and *P-charging-function-address* are deleted if present in messages received on this sip-interface. The enforcement-profile *filterHeadersCore* rejects unauthorized methods sent by the SSW to the SBC and filters out unnecessary SIP header by referring to a whitelist.

The option *strip-route-headers* removes any header '*Route*' from received requests (which would be honoured by the ESBC as described in RFC3261).

Element	Configuration
Core Realm	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# sip-interface CSBC(sip-interface)# realm-id Core CSBC(sip-interface)# charging-vector-mode delete CSBC(sip-interface)# charging-function-address-mode delete CSBC(sip-interface)# options +strip-route-headers CSBC(sip-interface)# enforcement-profile filterHeadersCore CSBC(sip-interface)# secured-network enabled CSBC(sip-interface)# response-map BT CSBC(sip-interface)# local-response-map localBT CSBC(sip-interface)# out-manipulationid outToBT CSBC(sip-interface)# sip-ports CSBC(sip-port)# address <SBC_CORE_IP> ex: 138.132.170.2 CSBC(sip-port)# port 5060 CSBC(sip-port)# allow-anonymous agents-only CSBC(sip-port)# done sip-interface state enabled realm-id Core description sip-port address 172.22.233.1 port 5060 transport-protocol UDP tls-profile allow-anonymous agents-only multi-home-addrs ims-aka-profile carriers trans-expire 0 initial-inv-trans-expire 0 invite-expire 0 session-max-life-limit 0 max-redirect-contacts 0 proxy-mode redirect-action contact-mode none nat-traversal none nat-interval 30 tcp-nat-interval 90 registration-caching disabled

```
min-reg-expire                  300
registration-interval           3600
route-to-registrar              disabled
secured-network                 disabled
teluri-scheme                   disabled
uri-fqdn-domain
options
spl-options
trust-mode                      all
max-nat-interval                3600
nat-int-increment               10
nat-test-increment              30
sip-dynamic-hnt                 disabled
stop-recurse                    401,407
port-map-start                  0
port-map-end                    0
in-manipulationid
out-manipulationid
sip-ims-feature                 disabled
sip-atcf-feature                disabled
subscribe-reg-event             disabled
operator-identifier
anonymous-priority              none
max-incoming-conns              0
per-src-ip-max-incoming-conns   0
inactive-conn-timeout           0
untrusted-conn-timeout          0
network-id
ext-policy-server
ldap-policy-server
default-location-string
term-tgrp-mode                  none
charging-vector-mode            pass
charging-function-address-mode  pass
ccf-address
ecf-address
implicit-service-route          disabled
rfc2833-payload                 101
rfc2833-mode                    transparent
constraint-name
response-map                    BT
local-response-map              localBT
sec-agree-feature               disabled
sec-agree-pref                  ipsec3gpp
enforcement-profile             filterHeadersCore
route-unauthorized-calls
tcp-keepalive                   none
add-sdp-invite                  disabled
add-sdp-in-msg
p-early-media-header            disabled
p-early-media-direction
add-sdp-profiles
add-sdp-profiles-in-msg
manipulation-string
manipulation-pattern
sip-profile
sip-isup-profile
tcp-conn-dereg                  0
tunnel-name
```

	register-keep-alive none kpml-interworking disabled kpml2833-iwf-on-hairpin disabled msrp-delay-egress-bye disabled send-380-response pcscf-restoration session-timer-profile session-recording-server session-recording-required disabled service-tag reg-cache-route disabled diversion-info-mapping-mode none atcf-icsi-match sip-recursion-policy asymmetric-preconditions disabled asymmetric-preconditions-mode send-with-nodelay sm-icsi-match-for-invite sm-icsi-match-for-message s8hr-profile ringback-trigger none ringback-file last-modified-by admin@10.253.51.48 last-modified-date 2019-09-18 15:26:27

1.2.1.3 Steering-pool Configuration

A single steering-pool must be provided for realm Core and shared by all Cisco IPBX, connected on Access South side, to exchange media on the core network.

The IP address is the same as the one used by core sip-interface.

A maximum set of 14000 ports can be configured to allow 7000 simultaneous calls depending of your BT/BTIP Voice channel orders.

Element	Configuration
Core Realm	CSBC# conf t CSBC(configure)# media-manager CSBC(media-manager)# steering-pool CSBC(steering-pool)# ip-address <SBC_CORE_IP> *ex: 138.132.170.2* CSBC(steering-pool)# start-port 6000 CSBC(steering-pool)# end-port 20000 CSBC(steering-pool)# realm-id Core CSBC(steering-pool)# done steering-pool ip-address 172.22.233.1 start-port 6000 end-port 20000 realm-id Core network-interface last-modified-by admin@10.253.51.48 last-modified-date 2019-04-02 16:01:56

1.2.2 Secured SIP Trunk through TLS

1.2.2.1 SBC Certfiicate

The SBC certificate is related to the sip-interface of the customer realm. The certificate-record is the object that holds the certificate. It must be created including:

- a common-name: (If FQDN used into Customer config)
- Country/state/locality: you can configure these fields with the SBC geographical location info.
- Organization/Unit: Customer name
- Extended-key-usage-list: the key will be used when the SBC acts both as client and server. So we must configure both serverAuth and clientAuth.
- A key size of 2048 bytes.

The certificate record name includes the creation date in order to track the various versions of the same certificate and facilitate the renewal.

Element	Configuration
Customer SBC certificates	CSBC# conf t CSBC (configure)# security certificate-record CSBC (certificate-record)# name CERT_ BTOI_<SBC_NAME>-<optionalSubName>_yyyymmdd CSBC (certificate-record)# done Warning: Required field "common-name" is empty Do you still want to save configuration [y/n]?: y certificate-record name CERT_BTOI_CSBC_ORACLE82_27082019 country FR state N/A locality Cesson_Sevigne organization Orange unit Orange business Services common-name CSBC key-size 2048 alternate-name trusted enabled key-usage-list digitalSignature keyEncipherment extended-key-usage-list serverAuth clientAuth key-algor rsa digest-algor sha256 ecdsa-key-size p256 cert-status-profile-list options last-modified-by admin@10.253.51.48 last-modified-date 2019-09-03 15:18:50 CSBC#done

Generate the CSR:

Customer SBC certificates	CSBC# generate-certificate-request CERT_BTOI_<SBC_NAME>_yyyymmdd Generating Certificate Signing Request. This can take several minutes.... -----BEGIN CERTIFICATE REQUEST----- MIIC+jCCAeICAQAweDELMAkGA1UEBhMCRlIxDDAKBgNVBAgTA04vQTEYMBYGA1UE BxQPQ2Vzc2lvbl9TZXZpZ25lMQ8wDQYDVQQKEwZPcmFuZ2UxITAfBgNVBAsTGE9y YW5nZSBidXNpbmVzcyBTZXJ2aWNlczENMAsGA1UEAxMEQ1NCQzCCASIwDQYJKoZI hvcNAQEBBQADggEPADCCAQoCggEBALK3Jqz99IYjLQa6MeD4IPGl3OhLQ5lkyIaB NRqQceh31nxpBMp6033n1RnG1Xc4DASK7DIiGWny55A3CvwKHWreC492my6PUT7D Zsl3w7jIYvos4KHBTZd+Z2RKdRzL1wwJvHnKWtdX+dq6ibVw9WimtQvIi3Qa3bS0 efQTzfSgx+9oTbe5RKatpW8UD9pEEqOxjU6kLH36D01IgSerPaR0EE0dfqtKBZIf AkessCbUePb+TgpNqpJ2JlstytvmZx2eS1w0NkdTkU872ntgFEj5UQh79/J5efLk s9KNHVYNGchwDOFnM7PIglSu720PpRpths2nL5YEmruSyM96yysCAwEAAaA9MDsG CSqGSIb3DQEJDjEuMCwwCwYDVR0PBAQDAgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMB BggrBgEFBQcDAjANBgkqhkiG9w0BAQsFAAOCAQEAYXnA7djcvExFkKreAKdTRNnq hJdtIJn8SjkLfmewiNOhFL/nau8NZs3aer75sBNt/KtfIbU3Onl9CoGh+bLajlxd c9fELKl5i4xQoNtRusBL5MoL30aVijfGRFcCaH48lrDynJ8iB4RL9gFyERwwGlPO NRdTL8ujtr9Hb6DlaeDP0G61+nePKvEp75ubhHIRImdciwTxXXL3cGxcSsxdR68C emG+iwAs7Q/rdJ6+RcqhK8bhV8LtekOeG+LVzmDWyoGadjSdVP77eqxTogzf+i1T pLNrSYt91nrMOOdTpVTbqdp3dDVOjF1itOSADKUgZg81ADi+y7v5ra5enW8cwg== -----END CERTIFICATE REQUEST----- WARNING: Configuration changed, run "save-config" command.

At this point you must save & activate. This is required so that the next step (generate the CSR) can be performed.

Customer SBC certificates	CSBC# save-config CSBC# activate-config

Then, obtain the CA signed certificate from the customer Certificate Autority in PKCS7 or X509v3 format encoded in PEM.

Import it using the copy/paste PEM content method or the sftp upload method od the PEM file

Customer SBC certificates	CSBC# import-certificate try-all CACERT_ BTOI_CSBC-<optionalSubName>_yyyymmdd Customer_SBC.pem Certificate imported successfully.... WARNING: Configuration changed, run "save-config" command. CSBC # save-config S CSBC # activate-config

You can now display the details of the signed certificate with the following CLI:

Customer SBC certificates	CSBC# show security certificates detail Certificate: Data: Version: 3 (0x2) Serial Number: 8 (0x8) Signature Algorithm: sha256WithRSAEncryption Issuer: C=FR ST=Bretagne L=Cesson O=Orange

	OU=Orange Business Services CN=CA_Test_SBC_OBS emailAddress=X.X@orange.com Validity Not Before: Sep 3 09:27:00 2019 GMT Not After : Sep 3 09:27:00 2020 GMT Subject: C=FR ST=N/A L=Cession_Sevigne O=Orange OU=Orange business Services CN=CSBC X509v3 extensions: X509v3 Key Usage: Digital Signature, Key Encipherment X509v3 Extended Key Usage: TLS Web Server Authentication, TLS Web Client Authentication

1.2.2.2 Customer CA certificate(s)

First we must configure the **certificate record that will hold the CA certificate.**

The record name includes a date of the certificate record creation. As it is possible that the customer provides several CA certificates, the record-name can include optionally a subname that helps to differentiates each one (the content of the subname can be freely chosen).
The procedure described in this chapter must be followed for each certificate.

Customer CA certificates	CSBC# security certificate-record CSBC # Name CACERT_< CUSTOMER_CA_NAME>_<optionalSubName>_yyyymmdd CSBC # done Warning: Required field "common-name" is empty Do you still want to save configuration [y/n]?: y certificate-record name CACERT_ORANGE_JLC_27082019 country FR state Bretagne locality Cesson organization Orange unit common-name key-size 1024 alternate-name trusted enabled key-usage-list digitalSignature keyEncipherment extended-key-usage-list serverAuth options last -modified-by admin@10.253.51.48 last-modified-date 2019-09-09 15:59:27

Then, obtain the CA signed certificate from the customer in PKCS7 or X509v3 format encoded in PEM. Import it using the copy/paste PEM content method or the sftp upload method od the PEM file

Customer SBC certificates	CSBC# import-certificate try-all CACERT_< CUSTOMER_CA_NAME>_<optionalSubName>_yyyymmdd Customer_CA.pem Certificate imported successfully.... WARNING: Configuration changed, run "save-config" command. CSBC # save-config S CSBC # activate-config

You can now display the details of the signed certificate with the following CLI:

Customer SBC certificates	CSBC# show security certificates detail certificate-record: CA_JLC_CA_09092019 Certificate: Data: Version: 3 (0x2) Serial Number: 1467967323148 (0x155c9abbc0c) Signature Algorithm: sha256WithRSAEncryption Issuer: C=FR O=Orange OU=OBS CN=JLC_CA Validity Not Before: Jul 8 08:42:03 2016 GMT Not After : Jul 9 08:42:16 2026 GMT Subject: C=FR O=Orange OU=OBS CN=JLC_CA X509v3 extensions: X509v3 Authority Key Identifier: keyid:47:7A:1F:F4:57:C0:7D:BF:0A:90:FA:23:F5:F3:CB:7D:75:C6:39:AE X509v3 Subject Key Identifier: 47:7A:1F:F4:57:C0:7D:BF:0A:90:FA:23:F5:F3:CB:7D:75:C6:39:AE X509v3 Key Usage: critical Digital Signature, Certificate Sign, CRL Sign X509v3 Basic Constraints: CA:TRUE

Indicates it's a CA certificate

1.2.2.3 TLS profile

Configure a TLS profile indicating the SBC certificate, the customer CA certificate(s), and the mutual authenticate method. If there are several customer certificates, they are entered within parenthesis, separated with a space.

BTOI TLS Profile	CSBC# conf t CSBC# security tls-profile CSBC# name tls-BTOI-profile CSBC# end-entity-certificate CERT_ BTOI_<SBC_NAME>-<optionalSubName>_yyyymmdd CSBC# trusted-ca-certificates CACERT_< CUSTOMER_CA_NAME>_<optionalSubName>_yyyymmdd CSBC# mutual-authenticate enabled CSBC#done tls-profile name tls-BTOI-profile end-entity-certificate CERT_BTOI_CSBC_ORACLE82_27082019 trusted-ca-certificates CA_JLC_CA_09092019 cipher-list all verify-depth 10 mutual-authenticate enabled tls-version tlsv12 options cert-status-check disabled cert-status-profile-list ignore-dead-responder disabled allow-self-signed-cert disabled last-modified-by admin@10.253.51.48 last-modified-date 2019-09-09 16:01:10

1.2.2.4 SRTP configuration

1.2.2.4.1 SDES profile

SDES is the key exchange protocol supported by the SBC for SRTP. The SBC is configured in single-ended SRTP termination mode (meaning the SBC terminate SRTP on access side, and use RTP on core side).
We define here a profile with AES/128 bit key for encryption and HMAC/SHA-1 80-bit digest for authentication, which is the default profile. RTP and RTCP are encrypted.

SDES profile	CSBC# conf t CSBC(configure)# security media-security sdes-profile CSBC(sdes-profile)# name SDES CSBC(sdes-profile)# crypto-list AES_CM_128_HMAC_SHA1_80 CSBC(sdes-profile)# done sdes-profile name SDES crypto-list AES_CM_128_HMAC_SHA1_80 srtp-auth enabled srtp-encrypt enabled srtcp-encrypt enabled mki disabled egress-offer-format same-as-ingress use-ingress-session-params options key salt

	srtp-rekey-on-re-invite disabled lifetime 0 last-modified-by admin@10.253.51.48 last-modified-date 2019-10-04 13:44:28

1.2.2.4.2 Media-sec-policy

This object defines the security policy to be applied on the media traffic. It calls the sdes profile defined in the previous paragraph.

Media-Sec-Policy	CSBC# conf t CSBC(configure)# security media-security media-sec-policy CSBC(media-sec-policy)# name msp-BTOI CSBC(media-sec-policy)# inbound CSBC(media-sec-inbound)# profile SDES CSBC(media-sec-inbound)# mode srtp CSBC(media-sec-inbound)# protocol sdes CSBC(media-sec-inbound)# done inbound profile SDES mode srtp protocol sdes CSBC(media-sec-inbound)# exit CSBC(media-sec-policy)# outbound CSBC(media-sec-outbound)# profile SDES CSBC(media-sec-outbound)# mode srtp CSBC(media-sec-outbound)# protocol sdes CSBC(media-sec-outbound)# done outbound profile SDES mode srtp protocol sdes CSBC(media-sec-outbound)# exit CSBC(media-sec-policy)# done media-sec-policy name msp-BTOI pass-through disabled options inbound profile SDES mode srtp protocol sdes hide-egress-media-update disabled outbound profile SDES mode srtp protocol sdes last-modified-by admin@10.253.51.48 last-modified-date 2019-10-04 13:47:25

1.2.2.5 Core realm Configuration

A core realm (id 'Core') is created once to represent the OBS carrier part of the ESBC and provides media-ports sharing. This realm is associated with a SIP-interface that is common for all Cisco iPBX.

When the customer uses SRTP, the media-sec-policy is configured with the value 'msp-BTOI'. When the customer uses RTP, the parameter is configured with the value 'nocrypto'.

<table>
<tr>
<td>BTOI TLS Profile</td>
<td>
<pre>
CSBC# conf t
CSBC(configure)# media-manager
CSBC(media-manager)# realm-config
CSBC(realm-config)# identifier Core
CSBC(realm-config)# network-interfaces M10:<VLAN>   ex: M10:187
CSBC(realm-config)# access-control-trust-level high
CSBC(realm-config)# mm-in-network enabled
CSBC(realm-config)# media-sec-policy msp-BTOI   if SRTP is used for
media
CSBC(realm-config)# media-sec-policy nocrypto   if RTP is used for media
CSBC(realm-config)# media-policy mark-mp
CSBC(realm-config)# codec-policy codecfiltering
CSBC(realm-config)# restricted-latching sdp
CSBC(realm-config)# done
realm-config
    identifier   Core
    description
    addr-prefix   0.0.0.0
    network-interfaces   M00:187
    mm-in-realm   disabled
    mm-in-network   enabled
    mm-same-ip   enabled
    mm-in-system   enabled
    bw-cac-non-mm   disabled
    msm-release   disabled
    qos-enable   disabled
    max-bandwidth   0
    fallback-bandwidth   0
    max-priority-bandwidth   0
    max-latency   0
    max-jitter   0
    max-packet-loss   0
    observ-window-size   0
    parent-realm
    dns-realm
    media-policy   mark-mp
    media-sec-policy   msp-BTOI
    rtcp-mux   disabled
    ice-profile
    dtls-srtp-profile
    srtp-msm-passthrough   disabled
    class-profile   mark-cp
    in-translationid
    out-translationid
    in-manipulationid
    out-manipulationid
    average-rate-limit   0
    access-control-trust-level   high
    invalid-signal-threshold   0
    maximum-signal-threshold   0
    untrusted-signal-threshold   0
</pre>
</td>
</tr>
</table>

```
nat-trust-threshold                 0
max-endpoints-per-nat               0
nat-invalid-message-threshold       0
wait-time-for-invalid-register      0
deny-period                         30
session-max-life-limit              0
cac-failure-threshold               0
untrust-cac-failure-threshold       0
ext-policy-svr
diam-e2-address-realm
subscription-id-type                END_USER_NONE
symmetric-latching                  disabled
pai-strip                           disabled
trunk-context                       -
device-id
early-media-allow
enforcement-profile
additional-prefixes
restricted-latching                 none
restriction-mask                    32
user-cac-mode                       none
user-cac-bandwidth                  0
user-cac-sessions                   0
icmp-detect-multiplier              0
icmp-advertisement-interval         0
icmp-target-ip
monthly-minutes                     0
options
spl-options
accounting-enable                   enabled
net-management-control              disabled
delay-media-update                  disabled
refer-call-transfer                 disabled
hold-refer-reinvite                 disabled
refer-notify-provisional            none
dyn-refer-term                      disabled
codec-policy
codec-manip-in-realm                disabled
codec-manip-in-network              enabled
rtcp-policy
constraint-name
session-recording-server
session-recording-required          disabled
manipulation-string
manipulation-pattern
stun-enable                         disabled
stun-server-ip                      0.0.0.0
stun-server-port                    3478
stun-changed-ip                     0.0.0.0
stun-changed-port                   3479
sip-profile
flow-time-limit                     -1
initial-guard-timer                 -1
subsq-guard-timer                   -1
tcp-flow-time-limit                 -1
tcp-initial-guard-timer             -1
tcp-subsq-guard-timer               -1
sip-isup-profile
match-media-profiles
```

	qos-constraint block-rtcp disabled hide-egress-media-update disabled tcp-media-profile monitoring-filters node-functionality default-location-string alt-family-realm pref-addr-type none sm-icsi-match-for-invite sm-icsi-match-for-message ringback-trigger none ringback-file last-modified-by admin@10.253.51.48 last-modified-date 2019-09-18 15:42:51

1.2.2.6 Core realm sip-interface

A sip-interface must be associated to the Core realm previously defined with the port 5061. Only one SIP interface is used on Core to represent all Cisco IPBX in front of BT/BTIP.

The parameter '*allow-anonymous agents-only*' enables only a provisioned session-agent to send requests to the ESBC: messages received from unknown sources will be rejected with '403 Forbidden'.

Headers *P-charging-vector* and *P-charging-function-address* are deleted if present in messages received on this sip-interface. The enforcement-profile *filterHeadersCore* rejects unauthorized methods sent by the SSW to the SBC and filters out unnecessary SIP header by referring to a whitelist.

The option *strip-route-headers* removes any header '*Route*' from received requests (which would be honoured by the ESBC as described in RFC3261).

Response-maps are called to map some SIP error codes. This is required in particular to enable re-routing by BT/BTIP infrastructure that needs to receive a 408 error code for that.

Element	Configuration

Core Realm Sip-interface	CSBC# conf t
	CSBC(configure)# session-router
	CSBC(session-router)# sip-interface
	CSBC(sip-interface)# realm-id Core
	CSBC(sip-interface)# charging-vector-mode delete
	CSBC(sip-interface)# charging-function-address-mode delete
	CSBC(sip-interface)# options +strip-route-headers
	CSBC(sip-interface)# enforcement-profile filterHeadersCore
	CSBC(sip-interface)# out-manipulationid outToBT
	CSBC(sip-interface)# stop-recurse 401-407
	CSBC(sip-interface)# secured-network enabled
	CSBC(sip-interface)# response-map BT
	CSBC(sip-interface)# local-response-map localBT
	CSBC(sip-interface)# sip-ports
	CSBC(sip-port)# address <SBC_CORE_IP> ex: 138.132.170.2
	CSBC(sip-port)# port 5061
	CSBC(sip-port)# allow-anonymous agents-only
	CSBC(sip-port)# transport-protocol TLS
	CSBC(sip-port)# tls-profile tls-BTOI-profile
	CSBC(sip-port)# exit
	CSBC(sip-interface)# done
	sip-interface
	    state enabled
	    realm-id Core
	    description
	    sip-port
	        address 172.22.233.1
	        port 5061
	        transport-protocol TLS
	        tls-profile tls-BTOI-profile
	        allow-anonymous agents-only
	        multi-home-addrs
	        ims-aka-profile
	    carriers
	    trans-expire 0
	    initial-inv-trans-expire 0
	    invite-expire 0
	    session-max-life-limit 0
	    max-redirect-contacts 0


```
proxy-mode
redirect-action
contact-mode                    none
nat-traversal                   none
nat-interval                    30
tcp-nat-interval                90
registration-caching            disabled
min-reg-expire                  300
registration-interval           3600
route-to-registrar              disabled
secured-network                 disabled
teluri-scheme                   disabled
uri-fqdn-domain
options
spl-options
trust-mode                      all
max-nat-interval                3600
nat-int-increment               10
nat-test-increment              30
sip-dynamic-hnt                 disabled
stop-recurse                    401,407
port-map-start                  0
port-map-end                    0
in-manipulationid
out-manipulationid
sip-ims-feature                 disabled
sip-atcf-feature                disabled
subscribe-reg-event             disabled
operator-identifier
anonymous-priority              none
max-incoming-conns              0
per-src-ip-max-incoming-conns   0
inactive-conn-timeout           0
untrusted-conn-timeout          0
network-id
ext-policy-server
ldap-policy-server
default-location-string
term-tgrp-mode                  none
charging-vector-mode            pass
charging-function-address-mode  pass
ccf-address
ecf-address
implicit-service-route          disabled
rfc2833-payload                 101
rfc2833-mode                    transparent
constraint-name
response-map                    BT
local-response-map              localBT
sec-agree-feature               disabled
sec-agree-pref                  ipsec3gpp
enforcement-profile             filterHeadersCore
route-unauthorized-calls
tcp-keepalive                   none
add-sdp-invite                  disabled
add-sdp-in-msg
p-early-media-header            disabled
p-early-media-direction
add-sdp-profiles
```

	add-sdp-profiles-in-msg manipulation-string manipulation-pattern sip-profile sip-isup-profile tcp-conn-dereg 0 tunnel-name register-keep-alive none kpml-interworking disabled kpml2833-iwf-on-hairpin disabled msrp-delay-egress-bye disabled send-380-response pcscf-restoration session-timer-profile session-recording-server session-recording-required disabled service-tag reg-cache-route disabled diversion-info-mapping-mode none atcf-icsi-match sip-recursion-policy asymmetric-preconditions disabled asymmetric-preconditions-mode send-with-nodelay sm-icsi-match-for-invite sm-icsi-match-for-message s8hr-profile ringback-trigger none ringback-file last-modified-by admin@10.253.51.48 last-modified-date 2019-09-18 15:26:27

1.2.2.1 Steering-pool Configuration

A single steering-pool must be provided for realm Core and shared by all Cisco IPBX, connected on Access South side, to exchange media on the core network.

The IP address is the same as the one used by core sip-interface.

A maximum set of 14000 ports can be configured to allow 7000 simultaneous calls depending of your BT/BTIP Voice channel orders.

Element	Configuration
Core Steering Pool	CSBC# conf t CSBC(configure)# media-manager CSBC(media-manager)# steering-pool CSBC(steering-pool)# ip-address <SBC_CORE_IP> *ex: 138.132.170.2* CSBC(steering-pool)# start-port 6000 CSBC(steering-pool)# end-port 20000 CSBC(steering-pool)# realm-id Core CSBC(steering-pool)# done steering-pool ip-address 172.22.233.1 start-port 6000 end-port 20000 realm-id Core network-interface last-modified-by admin@10.253.51.48 last-modified-date 2019-04-02 16:01:56

1.2.3 BT/BTIP objects

1.2.3.1 Nominal Session agent

A session-agent must be configured to represent each address by which BT/BTIP infrastructure can be targeted. The availability of any address is monitored through the periodic OPTIONS mechanism.

The session-agent is put out-of-service in case it doesn't answer a ping-transaction (OPTIONS sent every 180 sec) or it doesn't answer two subsequent non-ping transactions and will be put back in-service as soon as it start sending SIP traffic or it answers a ping-transaction.

Each address belongs either to the Nominal or the Backup group. If a call must be routed to BT/BTIP, a primary address will be chosen as target of the call, and then a secondary address in case all attempts to any element of the Nominal failed or none of the Nominal addresses is available or a specific error code is received which stops recursion.

For BT/BTIP will need at least the configuration of the first element of the Nominal group (never empty) while the Backup group might be empty which will depends of service contracted.

Element	Configuration
Main BT/BTIP session-agent	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# session-agent CSBC(session-agent)# hostname <BT_NOMINAL_SA > ex: BT_NOMINAL_SA or Public BT FQDN CSBC(session-agent)# ip-address <BT_NOMINAL_SA_IP> ex: 82.82.24.71 CSBC(session-agent)# port 5060 => For unsecured though UDP CSBC(session-agent)# port 5061 => For secured though TLS CSBC(session-agent)# transport-method UDP => For unsecured though UDP CSBC(session-agent)# transport-method StaticTLS => For secured though TLS CSBC(session-agent)# trust-me enabled CSBC(session-agent)# realm Core CSBC(session-agent)# ping-method OPTIONS CSBC(session-agent)# ping-interval 180 CSBC(session-agent)# constraints enabled CSBC(session-agent)# ttr-no-response 900 CSBC(session-agent)# options +trans-timeouts=2 CSBC(session-agent)# done session-agent hostname BT_NOMINAL_SA ip-address 172.22.246.33 port 5061 state enabled app-protocol SIP app-type transport-method StaticTLS => For secured though TLS transport-method UDP => For unsecured though UDP realm-id Core egress-realm-id description carriers allow-next-hop-lp enabled associated-agents constraints enabled max-sessions 0 max-inbound-sessions 0 max-outbound-sessions 0

```
max-burst-rate                 0
max-inbound-burst-rate         0
max-outbound-burst-rate        0
max-sustain-rate               0
max-inbound-sustain-rate       0
max-outbound-sustain-rate      0
min-seizures                   5
min-asr                        0
session-max-life-limit         0
time-to-resume                 0
ttr-no-response                900
in-service-period              0
burst-rate-window              0
sustain-rate-window            0
req-uri-carrier-mode           None
proxy-mode
redirect-action
loose-routing                  enabled
send-media-session             enabled
response-map
ping-method                    OPTIONS
ping-interval                  180
ping-send-mode                 keep-alive
ping-all-addresses             disabled
ping-in-service-response-codes
out-service-response-codes
load-balance-dns-query         hunt
options                        trans-timeouts=2
spl-options
media-profiles
in-translationid
out-translationid
trust-me                       enabled
request-uri-headers
stop-recurse
local-response-map
ping-to-user-part
ping-from-user-part
in-manipulationid
out-manipulationid
manipulation-string
manipulation-pattern
p-asserted-id
trunk-group
max-register-sustain-rate      0
early-media-allow
invalidate-registrations       disabled
rfc2833-mode                   none
rfc2833-payload                0
codec-policy
enforcement-profile
refer-call-transfer            disabled
refer-notify-provisional       none
reuse-connections              NONE
tcp-keepalive                  none
tcp-reconn-interval            0
max-register-burst-rate        0
register-burst-window          0
sip-profile
```

	sip-isup-profile kpml-interworking inherit kpml2833-iwf-on-hairpin inherit precedence 0 monitoring-filters session-recording-server session-recording-required disabled hold-refer-reinvite disabled send-tcp-fin disabled sip-recursion-policy sm-icsi-match-for-invite sm-icsi-match-for-message ringback-trigger none ringback-file last-modified-by admin@10.253.51.48 last-modified-date 2019-09-10 15:27:38

1.2.3.2 Backup Session Agent

Same for backup BT/BTIPSIP Termination mus be configured, please follow bellow :

Element	Configuration
Main BT/BTIP session-agent	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# session-agent CSBC(session-agent)# hostname <BT_BACKUP_SA > ex: BT_BACKUP_SA or Public BT FQDN CSBC(session-agent)# ip-address <BT_BACKUP_SA_IP> ex: 82.82.24.71 CSBC(session-agent)# port 5060 => For unsecured though UDP CSBC(session-agent)# port 5061 => For secured though TLS CSBC(session-agent)# transport-method UDP => For unsecured though UDP CSBC(session-agent)# transport-method StaticTLS => For secured though TLS CSBC(session-agent)# trust-me enabled CSBC(session-agent)# realm Core CSBC(session-agent)# ping-method OPTIONS CSBC(session-agent)# ping-interval 180 CSBC(session-agent)# constraints enabled CSBC(session-agent)# ttr-no-response 900 CSBC(session-agent)# options +trans-timeouts=2 CSBC(session-agent)# done session-agent hostname BT_BACKUP_SA ip-address 172.22.246.73 port 5061 state enabled

```
app-protocol                     SIP
app-type
transport-method                   StaticTLS => For secured though TLS
transport-method                   UDP => For unsecured though UDP
realm-id                         Core
egress-realm-id
description
carriers
allow-next-hop-lp                  enabled
associated-agents
constraints                      enabled
max-sessions                       0
max-inbound-sessions                 0
max-outbound-sessions                0
max-burst-rate                     0
max-inbound-burst-rate               0
max-outbound-burst-rate              0
max-sustain-rate                   0
max-inbound-sustain-rate             0
max-outbound-sustain-rate            0
min-seizures                      5
min-asr                          0
session-max-life-limit              0
time-to-resume                     0
ttr-no-response                    900
in-service-period                  0
burst-rate-window                   0
sustain-rate-window                  0
req-uri-carrier-mode                None
proxy-mode
redirect-action
loose-routing                     enabled
send-media-session                    enabled
response-map
ping-method                        OPTIONS
ping-interval                     180
ping-send-mode                      keep-alive
ping-all-addresses                 disabled
ping-in-service-response-codes
out-service-response-codes
load-balance-dns-query                 hunt
options                          trans-timeouts=2
spl-options
media-profiles
in-translationid
out-translationid
trust-me                         enabled
request-uri-headers
stop-recurse
local-response-map
ping-to-user-part
ping-from-user-part
in-manipulationid
out-manipulationid
manipulation-string
manipulation-pattern
p-asserted-id
trunk-group
max-register-sustain-rate              0
```

	early-media-allow invalidate-registrations disabled rfc2833-mode none rfc2833-payload 0 codec-policy enforcement-profile refer-call-transfer disabled refer-notify-provisional none reuse-connections NONE tcp-keepalive none tcp-reconn-interval 0 max-register-burst-rate 0 register-burst-window 0 sip-profile sip-isup-profile kpml-interworking inherit kpml2833-iwf-on-hairpin inherit precedence 0 monitoring-filters session-recording-server session-recording-required disabled hold-refer-reinvite disabled send-tcp-fin disabled sip-recursion-policy sm-icsi-match-for-invite sm-icsi-match-for-message ringback-trigger none ringback-file last-modified-by admin@10.253.51.48 last-modified-date 2019-09-10 15:27:38

1.2.3.3 Session Agent Groups

One groups need to be created for each BT/BTIP SIP termination,. They contain respectively the set of Nominal and Backup IP addresses of the BT/BTIP SIP termination.

Depending of the BT/BTIP architecture, a Nominal group and a Backup group can be configured

1.2.3.3.1 Nominal Session Agent Group

Element	Configuration
BT/BTIP Session Agent Group	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# session-group CSBC(session-agent-group)# group-name SSWCSBC CSBC(session-agent-group)# dest BT_NOMINAL_SA_IP CSBC(session-agent-group)# strategy hunt CSBC(session-agent-group)# sag-recursion enabled CSBC(session-agent-group)# stop-sag-recurse 400-407,409-499 CSBC(session-agent-group)# app-protocol SIP CSBC(session-agent-group)# done session-group group-name SSWCSBC description state enabled app-protocol SIP strategy Hunt dest SBC113 trunk-group sag-recursion enabled stop-sag-recurse 400-407,409-599 sip-recursion-policy last-modified-by admin@10.253.51.48 last-modified-date 2019-09-12 09:52:03

1.2.3.4 Access List

For each configured session-agent, an access-control is created specifying as source address the IP address of the session-agent, as destination-address the IP address of the sip-interface associated to the customer. A signaling packet whose source/destination don't match one of the configured access-controls will be discarded at IP level.

1.2.3.5 BT Nominal Session Agent- control

Element	Configuration
BT Nominal Session-Agent Access-Control	CSBC# conf t CSBC(configure)# session-router access-control CSBC(access-control)# source-address <BT_NOMINAL_SA_IP> ex: 82.82.24.71 CSBC(access-control)# destination-address <ESBC_NOMINAL_IP> ex: 138.132.169.2 CSBC(access-control)# realm-id Core CSBC(access-control)# application-protocol SIP CSBC(access-control)# access permit CSBC(access-control)# trust-level high CSBC(access-control)# transport-protocol UDP => For unsecured though UDP CSBC(access-control)# transport-protocol TCP => For secured though TLS CSBC(access-control)# done access-control realm-id Core description source-address 172.22.246.33 destination-address 172.22.233.1 application-protocol SIP transport-protocol TCP access permit average-rate-limit 0 trust-level high minimum-reserved-bandwidth 0 invalid-signal-threshold 0 maximum-signal-threshold 0 untrusted-signal-threshold 0 deny-period 30 nat-trust-threshold 0 max-endpoints-per-nat 0 nat-invalid-message-threshold 0 cac-failure-threshold 0 untrust-cac-failure-threshold 0 last-modified-by admin@10.253.51.48 last-modified-date 2019-09-12 13:22:48

1.2.3.6 BT Backup Session Agent- control

Element	Configuration
BT Backup Session-Agent Access-Control	CSBC# conf t CSBC(configure)# session-router access-control CSBC(access-control)# source-address <BT_BACKUP_SA_IP> ex: 82.82.24.71 CSBC(access-control)# destination-address <ESBC_NOMINAL_IP> ex: 138.132.169.2 CSBC(access-control)# realm-id Core CSBC(access-control)# application-protocol SIP CSBC(access-control)# access permit CSBC(access-control)# trust-level high CSBC(access-control)# transport-protocol UDP => For unsecured though UDP CSBC(access-control)# transport-protocol TCP => For secured though TLS CSBC(access-control)# done access-control realm-id Core description source-address 172.22.246.33 destination-address 172.22.233.1 application-protocol SIP transport-protocol TCP access permit average-rate-limit 0 trust-level high minimum-reserved-bandwidth 0 invalid-signal-threshold 0 maximum-signal-threshold 0 untrusted-signal-threshold 0 deny-period 30 nat-trust-threshold 0 max-endpoints-per-nat 0 nat-invalid-message-threshold 0 cac-failure-threshold 0 untrust-cac-failure-threshold 0 last-modified-by admin@10.253.51.48 last-modified-date 2019-09-12 13:22:48

1.2.4 Provisioning BT/BTIP on a backup ESBC

Perform exactly the same configuration as presented previously on the main SBC using parameters of backup SBC:

- <ESBC_SOUTH_BACKUP_GW>
- <ESBC_SOUTH_BACKUP_IP>

1.2.5 Local-policy from core to access

A local-policy must be created when a new customer Cisco CUCM IPBX is provisioned in order to route offnet calls from BT/ BTIP infrastructure towards correct customer Cisco IPBX.

The local-policy from core to access is provided with two next-hops: the Nominal group and the Backup group of the PBX. Note regardless the Backup group is empty, it will be put as alternative choice. The SBC will try first to route calls to the Nominal group, and only in case of failure (or all Nominal elements are out of service), a second routing attempt will be made to the Backup group. For seek of clearness a cost=1 is assigned to the route for the Backup group (being 0 for the Nominal group). The SBC chooses inside each group to which specific element the call has to be sent based on a round-robin strategy.

The next-hop for Cisco CUCM IPBX SIP is SSWCISCO (Session Agent Group for Cisco CUCM SIP terminations).

Element	Configuration

Local-policy from core to access	CSBC# conf t
	CSBC(configure)# session-router
	CSBC(session-router)# local-policy
	CSBC(local-policy)# from-address *
	CSBC(local-policy)# to-address (<4Digits started_range_DID> +<4Digits ended_range_DID + Private_Number) ex: (3329608 + 3329609 + 605)
	CSBC(local-policy)# source-realm Core
	CSBC(local-policy)# policy-attribute
	CSBC(local-policy-attributes)# next-hop SAG: N_<VLAN_ID>_<IPBX_VENDOR> ex: SAG:N_110_CISCO_CUCM
	CSBC(local-policy-attributes)# realm ACC_<VLAN_ID>_<IPBX_VENDOR> ex: ACC_110_CISCO_CUCM
	CSBC(local-policy-attributes)# app-protocol SIP
	CSBC(local-policy-attributes)# done
	local-policy
	from-address *
	to-address 3329608
	605
	+3329608
	source-realm Core
	description
	activate-time
	deactivate-time
	state enabled
	policy-priority none
	policy-attribute
	next-hop SAG:SSWCISCO
	realm ACC_331_CISCO_CUCM
	action none
	terminate-recursion disabled
	carrier
	start-time 0000
	end-time 2400
	days-of-week U-S
	cost 0
	state enabled
	app-protocol SIP
	methods
	media-profiles
	lookup single

	next-key eloc-str-lkup disabled eloc-str-match last-modified-by admin@10.253.51.48 last-modified-date 2019-09-12 12:23:13 CSBC(local-policy-attributes)# next-hop SAG: B_<VLAN_ID>_<IPBX_VENDOR> ex: SAG:B_110_CISCO_CUCM CSBC(local-policy-attributes)# realm ACC_<VLAN_ID>_<IPBX_VENDOR> ex: ACC_110_CISCO_CUCM CSBC(local-policy-attributes)# cost 1 CSBC(local-policy-attributes)# app-protocol SIP CSBC(local-policy-attributes)# done policy-attribute next-hop SAG:SSWCISCO realm ACC_331_CISCO_CUCM action none terminate-recursion disabled carrier start-time 0000 end-time 2400 days-of-week U-S cost 1 app-protocol SIP state enabled media-profiles

```
 CSBC(configure)# session-router
 CSBC(session-router)# local-policy
 CSBC(local-policy)# from-address *
 CSBC(local-policy)# to-address (04<T1T7> +04<T1T7>)
 CSBC(local-policy)# source-realm Core
 CSBC(local-policy)# policy-attribute
 CSBC(local-policy-attributes)# next-hop
SAG:N_<VLAN_ID>_<IPBX_VENDOR>_<T1T7>_<SIP_PROFILE>   ex: SAG:N_110_orange_1234567_01
 CSBC(local-policy-attributes)# realm ACC_<VLAN_ID>_<IPBX_VENDOR>    ex:
ACC_110_orange
 CSBC(local-policy-attributes)# app-protocol SIP
 CSBC(local-policy-attributes)# done

        policy-attribute
                next-hop                       SAG:N_110_orange_1234567_01
                realm                          ACC_110_orange
                action                         none
                terminate-recursion            disabled
                carrier
                start-time                     0000
                end-time                       2400
                days-of-week                   U-S
                cost                           0
                app-protocol                   SIP
                state                          enabled
                media-profiles

 CSBC(local-policy-attributes)# next-hop
SAG:B_<VLAN_ID>_<IPBX_VENDOR>_<T1T7>_<SIP_PROFILE>   ex: SAG:B_110_orange_1234567_01
 CSBC(local-policy-attributes)# realm ACC_<VLAN_ID>_<IPBX_VENDOR>    ex:
ACC_110_orange
CSBC(local-policy-attributes)# cost 1
CSBC(local-policy-attributes)# app-protocol SIP
CSBC(local-policy-attributes)# done

        policy-attribute
```



```
                next-hop                        SAG:B_110_orange_1234567_01
                realm                           ACC_110_orange
                action                          none
                terminate-recursion             disabled
                carrier
                start-time                      0000
                end-time                        2400
                days-of-week                    U-S
                cost                            1
                app-protocol                    SIP
                state                           enabled
                media-profiles

 CSBC(local-policy-attributes)# exit
 CSBC(local-policy)# done

local-policy
        from-address
                                        *
        to-address
                                        041234567 +041234567
        source-realm
                                        Core
        activate-time                   N/A
        deactivate-time                 N/A
        state                           enabled
        policy-priority                 none
        last-modified-date              2007-11-09 13:59:29
        policy-attribute
                next-hop                        SAG:N_110_orange_1234567_01
                realm                           ACC_110_orange
                action                          none
                terminate-recursion             disabled
                carrier
                start-time                      0000
                end-time                        2400
                days-of-week                    U-S
                cost                            0
                app-protocol                    SIP
                state                           enabled
                media-profiles
        policy-attribute
                next-hop                        SAG:B_110_orange_1234567_01
                realm                           ACC_110_orange
                terminate-recursion             disabled
                carrier
                start-time                      0000
                end-time                        2400
                days-of-week                    U-S
                cost                            1
                app-protocol                    SIP
                state                           enabled
                media-profiles

 CSBC(local-policy)# exit
 CSBC(session-router)# exit
 CSBC(configure)#
```

1.3 Customer Cisco CUCM IPBX South SIP configuration for Oracle SBC configuration

1.3.1 Provisioning a Cisco CUCM IPBX on the ESBC

Adding the configuration related to a Cisco CUCM IPBX involves the creation of all the configuration objects which are referred by the new objects to be provisioned are:

- the session-agents representing all the addresses associated to the PBX we are provisioning.
- One access-control for each configured session-agent, in order to discard traffic received from unknown sources.
- the Nominal and the Backup session groups into which the session-agents are partitioned.
- one local-policy from core to access to route incoming SIP calls to the Cisco CUCM IPBX

1.3.1.1 Access Network interface

Create the access network-interface associated to a Cisco CUCM IPBX as shown below. The sub-port-id parameter indicates the VLAN tag of the network.

Note that the IP address of the network-interface will be used for sending/receiving signalling (sip-interface) and media (steering-pool) related to this Cisco CUCM IPBX.

The hip-ip-list and icmp-address parameters are left empty in order to disable ping on the media interfaces. In order to enable temporarily the ping on the media interfaces for troubleshooting reason, you can add ip-address configured into.

Element	Configuration

Access Network interface	CSBC# conf t
	CSBC(configure)# system
	CSBC(system)# network-interface
	CSBC(network-interface)# name M10
	CSBC(network-interface)# sub-port-id <VLAN_ID> ex: 110
	CSBC(network-interface)# ip-address <ESBC_SOUTH_NOMINAL_IP> *ex: 138.132.169.2*
	CSBC(network-interface)# netmask 255.255.255.248
	CSBC(network-interface)# gateway <ESBC_SOUTH_NOMINAL_GW> *ex: 138.132.169.1*
	CSBC(network-interface)# done
	network-interface
	    name M10
	    sub-port-id 331
	    description
	    hostname
	    ip-address 6.6.5.1
	    pri-utility-addr
	    sec-utility-addr
	    netmask 255.255.255.0
	    gateway 6.6.5.254
	    sec-gateway
	    gw-heartbeat
	        state disabled
	        heartbeat 0
	        retry-count 0
	        retry-timeout 1
	        health-score 0
	    bfd-config

	state disabled health-score 0 options dns-ip-primary dns-ip-backup1 dns-ip-backup2 dns-domain dns-timeout 11 dns-max-ttl 86400 signaling-mtu 0 hip-ip-list 6.6.5.1 icmp-address 6.6.5.1 snmp-address ssh-address last-modified-by admin@6.0.3.100 last-modified-date 2019-03-21 15:36:50 CSBC#

1.3.1.2 Access Realm

An access realm must be created whenever a new customer CUCM IPBXs is provisioned to allow sharing the media and signalling ports on the SBC by all the CUCM IPBXs of the customer.

For SIP profile controls reinforcements, the codec-policy is now configured in order to filter out unsupported audio codec and media types.

Element	Configuration
Access Network interface	CSBC# conf t CSBC(configure)# media-manager CSBC(media-manager)# realm-config CSBC(realm-config)# identifier ACC_<VLAN_ID>_<IPBX_VENDOR> ex: ACC_110_CISCO_CUCM CSBC(realm-config)# network-interfaces M10:<VLAN_ID> ex: M10:110 CSBC(realm-config)# access-control-trust-level high CSBC(realm-config)# media-policy mark-mp CSBC(realm-config)# class-profile mark-cp CSBC(realm-config)# mm-in-network disabled CSBC(realm-config)# restricted-latching sdp CSBC(realm-config)# trunk-context <VLAN_ID> CSBC(realm-config)# codec-policy codecfiltering CSBC(realm-config)# done realm-config identifier ACC_331_CISCO_CUCM description addr-prefix 0.0.0.0 network-interfaces M10:331 mm-in-realm disabled mm-in-network enabled mm-same-ip enabled mm-in-system enabled bw-cac-non-mm disabled msm-release disabled qos-enable disabled max-bandwidth 0 fallback-bandwidth 0 max-priority-bandwidth 0 max-latency 0

```
max-jitter                        0
max-packet-loss                   0
observ-window-size                0
parent-realm
dns-realm
media-policy                      mark-mp
media-sec-policy                  nocrypto
rtcp-mux                          disabled
ice-profile
dtls-srtp-profile
srtp-msm-passthrough              disabled
class-profile                     mark-cp
in-translationid
out-translationid
in-manipulationid
out-manipulationid                outToPBXsipManip
average-rate-limit                0
access-control-trust-level        high
invalid-signal-threshold          0
maximum-signal-threshold          0
untrusted-signal-threshold        0
nat-trust-threshold               0
max-endpoints-per-nat             0
nat-invalid-message-threshold     0
wait-time-for-invalid-register    0
deny-period                       30
session-max-life-limit            0
cac-failure-threshold             0
untrust-cac-failure-threshold     0
ext-policy-svr
diam-e2-address-realm
subscription-id-type              END_USER_NONE
symmetric-latching                disabled
pai-strip                         disabled
trunk-context
device-id
early-media-allow
enforcement-profile
additional-prefixes
restricted-latching               none
restriction-mask                  32
user-cac-mode                     none
user-cac-bandwidth                0
user-cac-sessions                 0
icmp-detect-multiplier            0
icmp-advertisement-interval       0
icmp-target-ip
monthly-minutes                   0
options
spl-options
accounting-enable                 enabled
net-management-control            disabled
delay-media-update                enabled
refer-call-transfer               disabled
hold-refer-reinvite               disabled
refer-notify-provisional          none
dyn-refer-term                    disabled
codec-policy                      codecfiltering
codec-manip-in-realm              disabled
```

	codec-manip-in-network enabled rtcp-policy constraint-name session-recording-server session-recording-required disabled manipulation-string manipulation-pattern stun-enable disabled stun-server-ip 0.0.0.0 stun-server-port 3478 stun-changed-ip 0.0.0.0 stun-changed-port 3479 sip-profile flow-time-limit -1 initial-guard-timer -1 subsq-guard-timer -1 tcp-flow-time-limit -1 tcp-initial-guard-timer -1 tcp-subsq-guard-timer -1 sip-isup-profile match-media-profiles qos-constraint block-rtcp disabled hide-egress-media-update disabled tcp-media-profile monitoring-filters node-functionality default-location-string alt-family-realm pref-addr-type none sm-icsi-match-for-invite sm-icsi-match-for-message ringback-trigger none ringback-file last-modified-by admin@10.253.51.48 last-modified-date 2019-09-10 16:14:06

1.3.1.3 Access Steering-pool

The same steering-pool will be used by the SBC to exchange media traffic with all PBXs belonging to this customer.

As example 14000 UDP ports are required to manage 7000 simultaneous calls, feel free to adapt those to your context.

Element	Configuration
Access Steering-pool	CSBC# conf t CSBC(configure)# media-manager CSBC(media-manager)# steering-pool CSBC(steering-pool)# ip-address <ESBC_SOUTH_NOMINAL_IP> ex: 138.132.169.2 CSBC(steering-pool)# start-port 6000 CSBC(steering-pool)# end-port 20000 CSBC(steering-pool)# realm-id ACC_<VLAN_ID>_<IPBX_VENDOR> ex: ACC_110_CUCM CSBC(steering-pool)# done steering-pool ip-address 172.22.233.1 start-port 6000 end-port 20000 realm-id Core network-interface last-modified-by admin@10.253.51.48 last-modified-date 2019-04-02 16:01:56

1.3.1.4 Access sip-interface

A new access sip-interface must be created when a new customer Cisco CUCM IPBX is provisioned. The sip-interface defines the socket that will be used by the SBC to exchange signalling with all this customer's Cisco PBXs.

Headers *P-charging-vector* and *P-charging-function-address* are deleted if present in messages received on this sip-interface. The enforcement-profile *filtermsg* rejects unauthorized methods sent by a PBX to the SBC and filters out unnecessary SIP headers. The option *strip-route-headers* removes any header '*Route*' from received requests (which would be honoured by the SBC as described in RFC3261). The inbound sip-manipulation has the purpose of removing undesired headers or modifying them in messages received by any PBX before being elaborated by the SBC.

The parameter *'allow-anonymous agents-only'* enables only a provisioned PBX to send request to the SBC: messages received from unknown sources will be rejected with '403 Forbidden'.

Element	Configuration
Access sip-interface	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# sip-interface CSBC(sip-interface)# realm-id ACC_<VLAN_ID>_<IPBX_VENDOR> ex: ACC_110_orange CSBC(sip-interface)# charging-vector-mode delete CSBC(sip-interface)# charging-function-address-mode delete CSBC(sip-interface)# options +strip-route-headers CSBC(sip-interface)# enforcement-profile filtermsg CSBC(sip-interface)# secured-network enabled CSBC(sip-interface)# local-response-map BT CSBC(sip-interface)# sip-ports CSBC(sip-port)# address <ESBC_SOUTH_NOMINAL_IP> ex: 138.132.169.2 CSBC(sip-port)# allow-anonymous agents-only CSBC(sip-port)# exit CSBC(sip-interface)# done sip-interface state enabled realm-id ACC_331_CISCO_WARZAW description sip-port address 6.6.5.1 port 5060 transport-protocol UDP tls-profile allow-anonymous all multi-home-addrs ims-aka-profile carriers trans-expire 0 initial-inv-trans-expire 0 invite-expire 0 session-max-life-limit 0 max-redirect-contacts 0 proxy-mode redirect-action contact-mode none nat-traversal none nat-interval 30 tcp-nat-interval 90 registration-caching disabled min-reg-expire 300 registration-interval 3600 route-to-registrar disabled secured-network enabled teluri-scheme disabled uri-fqdn-domain options strip-route-headers spl-options trust-mode all max-nat-interval 3600 nat-int-increment 10 nat-test-increment 30 sip-dynamic-hnt disabled stop-recurse 401,407 port-map-start 0

	port-map-end 0 in-manipulationid manip-in-fromPBX out-manipulationid manip-out-toPBX sip-ims-feature disabled sip-atcf-feature disabled subscribe-reg-event disabled operator-identifier anonymous-priority none max-incoming-conns 0 per-src-ip-max-incoming-conns 0 inactive-conn-timeout 0 untrusted-conn-timeout 0 network-id ext-policy-server ldap-policy-server default-location-string term-tgrp-mode none charging-vector-mode delete charging-function-address-mode delete ccf-address ecf-address implicit-service-route disabled rfc2833-payload 101 rfc2833-mode transparent constraint-name response-map local-response-map BT sec-agree-feature disabled sec-agree-pref ipsec3gpp enforcement-profile filtermsg route-unauthorized-calls tcp-keepalive none add-sdp-invite disabled add-sdp-in-msg p-early-media-header disabled p-early-media-direction add-sdp-profiles add-sdp-profiles-in-msg manipulation-string manipulation-pattern sip-profile sip-isup-profile tcp-conn-dereg 0 tunnel-name register-keep-alive none kpml-interworking disabled kpml2833-iwf-on-hairpin disabled msrp-delay-egress-bye disabled send-380-response pcscf-restoration session-timer-profile session-recording-server session-recording-required disabled service-tag reg-cache-route disabled diversion-info-mapping-mode none atcf-icsi-match sip-recursion-policy asymmetric-preconditions disabled

	asymmetric-preconditions-mode send-with-nodelay sm-icsi-match-for-invite sm-icsi-match-for-message s8hr-profile ringback-trigger none ringback-file last-modified-by admin@10.253.51.48 last-modified-date 2019-09-18 15:07:47

1.3.2 Provisioning a new customer Cisco IPBX on a backup ESBC

Perform exactly the same configuration as presented previously on the backup SBC using parameters of backup SBC:

- <ESBC_SOUTH_BACKUP_GW>
- <ESBC_SOUTH_BACKUP_IP>

1.3.3 Cisco IPBX objects

1.3.3.1 Nominal Session agent

A session-agent must be configured to represent each address by which the Cisco IPBX can be targeted. The availability of any address is monitored through the periodic OPTIONS mechanism.

The session-agent is put out-of-service in case it doesn't answer a ping-transaction (OPTIONS sent every 180 sec) or it doesn't answer two subsequent non-ping transactions and will be put back in-service as soon as it start sending SIP traffic or it answers a ping-transaction.

Each address belongs either to the Nominal or the Backup group. If a call must be routed to the IPBX, a primary address will be chosen as target of the call, and then a secondary address in case all attempts to any element of the Nominal failed or none of the Nominal addresses is available or a specific error code is received which stops recursion.

A Cisco IPBX will need at least the configuration of the first element of the Nominal group (never empty) while the Backup group might be empty.

Element	Configuration
Main Access session-agent	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# session-agent CSBC(session-agent)# hostname N-<IPBX_VLAN>-<IPBX_VENDOR>-<SA_X> ex: N-331-CISCO-CUCM-SA-01 CSBC(session-agent)# ip-address <IPBX_NOMINAL_SA_IP> ex: 82.82.24.71 CSBC(session-agent)# port 5060 CSBC(session-agent)# trust-me enabled CSBC(session-agent)# realm ACC_<IPBX_VLAN_ID>_<IPBX_VENDOR> ex: ACC_331_CISCO_CUCM CSBC(session-agent)# ping-method OPTIONS CSBC(session-agent)# ping-interval 180 CSBC(session-agent)# constraints enabled CSBC(session-agent)# ttr-no-response 900 CSBC(session-agent)# options +trans-timeouts=2

```
CSBC(session-agent)# done
session-agent
        hostname                        N-331-CISCO-CUCM-SA-01
        ip-address                      6.5.6.1
        port                            5060
        state                           enabled
        app-protocol                    SIP
        app-type
        transport-method                UDP
        realm-id                        ACC_331_CISCO_CUCM
        egress-realm-id
        description                     Nominal_CUCM_CUCM
        carriers
        allow-next-hop-lp               enabled
        associated-agents
        constraints                     enabled
        max-sessions                    0
        max-inbound-sessions            0
        max-outbound-sessions           0
        max-burst-rate                  0
        max-inbound-burst-rate          0
        max-outbound-burst-rate         0
        max-sustain-rate                0
        max-inbound-sustain-rate        0
        max-outbound-sustain-rate       0
        min-seizures                    5
        min-asr                         0
        session-max-life-limit          0
        time-to-resume                  0
        ttr-no-response                 900
        in-service-period               0
        burst-rate-window               0
        sustain-rate-window             0
        req-uri-carrier-mode            None
        proxy-mode
        redirect-action
        loose-routing                   enabled
        send-media-session              enabled
        response-map
        ping-method                     OPTIONS
        ping-interval                   180
        ping-send-mode                  keep-alive
        ping-all-addresses              disabled
        ping-in-service-response-codes
        out-service-response-codes
        load-balance-dns-query          hunt
        options                         trans-timeouts=2
        spl-options
        media-profiles
        in-translationid
        out-translationid
        trust-me                        enabled
        request-uri-headers
        stop-recurse
        local-response-map
        ping-to-user-part
        ping-from-user-part
        in-manipulationid
        out-manipulationid
```

	manipulation-string manipulation-pattern p-asserted-id trunk-group max-register-sustain-rate 0 early-media-allow invalidate-registrations disabled rfc2833-mode none rfc2833-payload 0 codec-policy enforcement-profile refer-call-transfer disabled refer-notify-provisional none reuse-connections NONE tcp-keepalive none tcp-reconn-interval 0 max-register-burst-rate 0 register-burst-window 0 sip-profile sip-isup-profile kpml-interworking inherit kpml2833-iwf-on-hairpin inherit precedence 0 monitoring-filters session-recording-server session-recording-required disabled hold-refer-reinvite disabled send-tcp-fin disabled sip-recursion-policy sm-icsi-match-for-invite sm-icsi-match-for-message ringback-trigger none ringback-file last-modified-by admin@6.0.3.245 last-modified-date 2019-03-26 10:29:35

1.3.3.2 Backup Session Agent

If any backup Cisco CUCM SIP Termination mus be configured, please follow bellow :

Element	Configuration
Backup Access session-agent	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# session-agent CSBC(session-agent)# hostname B-<IPBX_VLAN>-<IPBX_VENDOR>-<SA_X> *ex: B-331-CISCO-CUCM-SA-01* CSBC(session-agent)# ip-address <IPBX_BACKUP_SA_IP> *ex: 82.82.24.71* CSBC(session-agent)# port 5060 CSBC(session-agent)# trust-me enabled CSBC(session-agent)# realm ACC_<IPBX_VLAN_ID>_<IPBX_VENDOR> ex: ACC_331_CISCO_CUCM CSBC(session-agent)# ping-method OPTIONS CSBC(session-agent)# ping-interval 180 CSBC(session-agent)# constraints enabled CSBC(session-agent)# ttr-no-response 900 CSBC(session-agent)# options +trans-timeouts=2 CSBC(session-agent)# done


```
session-agent
        hostname                        B-331-CISCO-WARZAW-SA-01
        ip-address                      6.5.6.2
        port                            5060
        state                           enabled
        app-protocol                    SIP
        app-type
        transport-method                UDP
        realm-id                        ACC_331_CISCO_WARZAW
        egress-realm-id
        description                     backup_CUCM_WARZAW
        carriers
        allow-next-hop-lp               enabled
        associated-agents
        constraints                     disabled
        max-sessions                    0
        max-inbound-sessions            0
        max-outbound-sessions           0
        max-burst-rate                  0
        max-inbound-burst-rate          0
        max-outbound-burst-rate         0
        max-sustain-rate                0
        max-inbound-sustain-rate        0
        max-outbound-sustain-rate       0
        min-seizures                    5
        min-asr                         0
        session-max-life-limit          0
        time-to-resume                  0
        ttr-no-response                 0
        in-service-period               0
        burst-rate-window               0
        sustain-rate-window             0
        req-uri-carrier-mode            None
        proxy-mode
        redirect-action
        loose-routing                   enabled
        send-media-session              enabled
        response-map
        ping-method                     OPTIONS
        ping-interval                   180
        ping-send-mode                  keep-alive
        ping-all-addresses              disabled
        ping-in-service-response-codes
        out-service-response-codes
        load-balance-dns-query          hunt
        options                         trans-timeouts=2
        spl-options
        media-profiles
        in-translationid
        out-translationid
        trust-me                        enabled
        request-uri-headers
        stop-recurse
        local-response-map
        ping-to-user-part
        ping-from-user-part
        in-manipulationid
        out-manipulationid
        manipulation-string
```

	manipulation-pattern p-asserted-id trunk-group max-register-sustain-rate 0 early-media-allow invalidate-registrations disabled rfc2833-mode none rfc2833-payload 0 codec-policy enforcement-profile refer-call-transfer disabled refer-notify-provisional none reuse-connections NONE tcp-keepalive none tcp-reconn-interval 0 max-register-burst-rate 0 register-burst-window 0 sip-profile sip-isup-profile kpml-interworking inherit kpml2833-iwf-on-hairpin inherit precedence 0 monitoring-filters session-recording-server session-recording-required disabled hold-refer-reinvite disabled send-tcp-fin disabled sip-recursion-policy sm-icsi-match-for-invite sm-icsi-match-for-message ringback-trigger none ringback-file last-modified-by admin@6.0.3.246 last-modified-date 2019-04-16 09:33:13

1.3.3.3 Session Agent Groups

Two groups need to be created for each Cisco CUCM IPBX SIP termination, the Nominal group and the Backup group. They contain respectively the set of Nominal and Backup IP addresses of the CUCM IPBX SIP termination.

To ease provisioning tasks, the backup group is always created even if it is left empty when there is no backup PBX.

Here we assume that each group is declared with one element.

1.3.3.3.1 Nominal Session Agent Group

Element	Configuration
Nominal Session Agent Group	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# session-group CSBC(session-agent-group)# group-name SSWCISCO CSBC(session-agent-group)# dest +N-<VLAN_ID>-<IPBX_VENDOR-<SA_X> ex: +N-331- CISCO_CUCM -01 CSBC(session-agent-group)# strategy roundrobin CSBC(session-agent-group)# sag-recursion enabled CSBC(session-agent-group)# stop-sag-recurse 400-407,409-499 CSBC(session-agent-group)# app-protocol SIP CSBC(session-agent-group)# done session-group group-name SSWCISCO description state enabled app-protocol SIP strategy Hunt dest N-331-CISCO-CUCM-SA-01 N-331-CISCO-CUCM-SA-02 trunk-group sag-recursion enabled stop-sag-recurse 400-407,409-599 sip-recursion-policy last-modified-by admin@6.0.3.245 last-modified-date 2019-03-28 17:10:25

1.3.3.3.2 Backup Session Agent Group

Element	Configuration
Backup Session Agent Group	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# session-group CSBC(session-agent-group)# group-name B_<VLAN_ID>_<IPBX_VENDOR> ex: B_331_CISCO_CUCM CSBC(session-agent-group)# dest +B-<VLAN_ID>-<IPBX_VENDOR-<SA_X> ex: +B-331- CISCO_CUCM -01 CSBC(session-agent-group)# strategy roundrobin CSBC(session-agent-group)# sag-recursion enabled CSBC(session-agent-group)# stop-sag-recurse 400-407,409-499 CSBC(session-agent-group)# app-protocol SIP CSBC(session-agent-group)# done

1.3.3.4 Access List

For each configured session-agent, an access-control is created specifying as source address the IP address of the session-agent, as destination-address the IP address of the sip-interface associated to the customer ESBC. A signaling packet whose source/destination don't match one of the configured access-controls will be discarded at IP level.

1.3.3.5 PBX Nominal Session Agent- control

Element	Configuration
PBX Nominal Session-Agent Access-Control	CSBC# conf t CSBC(configure)# session-router access-control CSBC(access-control)# source-address <IPBX_NOMINAL_SA_IP> ex: 82.82.24.71 CSBC(access-control)# destination-address <SBC_NOMINAL_IP> ex: 138.132.169.2 CSBC(access-control)# realm-id ACC_<VLAN_ID>_<IPBX_VENDOR> ex: ACC_110_orange CSBC(access-control)# application-protocol SIP CSBC(access-control)# access permit CSBC(access-control)# trust-level high CSBC(access-control)# transport-protocol UDP CSBC(access-control)# done access-control realm-id ACC_331_CISCO_CUCM description source-address 6.5.6.1 destination-address 6.6.5.1 application-protocol SIP transport-protocol ALL access permit average-rate-limit 0 trust-level high minimum-reserved-bandwidth 0 invalid-signal-threshold 0 maximum-signal-threshold 0 untrusted-signal-threshold 0 deny-period 30 nat-trust-threshold 0 max-endpoints-per-nat 0 nat-invalid-message-threshold 0 cac-failure-threshold 0 untrust-cac-failure-threshold 0 last-modified-by admin@10.253.51.48 last-modified-date 2019-09-10 08:09:50

1.3.3.6 PBX Backup Session Agent- control

Element	Configuration
PBX Nominal Session-Agent Access-Control	CSBC# conf t CSBC(configure)# session-router access-control CSBC(access-control)# source-address <IPBX_BACKUP_SA_IP> ex: 82.82.24.71 CSBC(access-control)# destination-address <SBC_NOMINAL_IP> ex: 138.132.169.2 CSBC(access-control)# realm-id ACC_<VLAN_ID>_<IPBX_VENDOR> ex: ACC_110_orange CSBC(access-control)# application-protocol SIP CSBC(access-control)# access permit CSBC(access-control)# trust-level high CSBC(access-control)# transport-protocol UDP CSBC(access-control)# done access-control realm-id ACC_331_CISCO_CUCM description source-address 6.5.6.2 destination-address 6.6.5.1 application-protocol SIP transport-protocol ALL access permit average-rate-limit 0 trust-level none minimum-reserved-bandwidth 0 invalid-signal-threshold 0 maximum-signal-threshold 0 untrusted-signal-threshold 0 deny-period 30 nat-trust-threshold 0 max-endpoints-per-nat 0 nat-invalid-message-threshold 0 cac-failure-threshold 0 untrust-cac-failure-threshold 0 last-modified-by admin@10.253.51.48 last-modified-date 2019-09-10 08:19:29

1.3.4 Local-policy from access to core

A local-policy must be created when a new customer Cisco CUCM IPBX is provisioned in order to route all calls from that customer Cisco IPBX towards the correct BT/ BTIP infratructure.

The local-policy from access to core is made of a single next-hop which is the group in which both the primary and the secondary SSWs are included. Also here, the SBC chooses which specific element the call has to be sent to based on a *hunt* strategy.

The next-hop for BTIP/BT SIP is SSW (Session Agent Group for BTIP SBCs).

Element	Configuration
Access sip-interface	CSBC# conf t CSBC(configure)# session-router CSBC(session-router)# local-policy CSBC(local-policy)# from-address * CSBC(local-policy)# to-address * CSBC(local-policy)# source-realm ACC_<VLAN_ID>_<IPBX_VENDOR> *ex: ACC_110_CISCO_CUCM* CSBC(local-policy)# policy-attribute CSBC(local-policy-attributes)# next-hop SAG:SSW for BTIP/BT SIP CSBC(local-policy-attributes)# realm Core CSBC(local-policy-attributes)# app-protocol SIP CSBC(local-policy-attributes)# done CSBC(local-policy-attributes)# exit CSBC(local-policy)# done local-policy from-address * to-address * source-realm ACC_331_CISCO_WARZAW description activate-time deactivate-time state enabled policy-priority none policy-attribute next-hop SAG:SSWCSBC realm Core action none terminate-recursion disabled carrier start-time 0000 end-time 2400 days-of-week U-S cost 0 state enabled app-protocol SIP methods media-profiles lookup single next-key eloc-str-lkup disabled eloc-str-match last-modified-by admin@10.253.51.48 last-modified-date 2019-04-02 15:47:26

1.4 SIP manipulations

Several SIP manipulations (aka "HMR") are required to manipulate the SIP headers and the sdp body, in order to control the content of the messages, and ensure the interoperability with the BTIP/BT services.

The SIP manipulations are provided as gzipped files. They are imported in the SBC configuration so that it is not necessary anymore to enter all the CLIs required to define each HMR. Only the CLI to import the files and the resulting HMR content are described in the document.

The HMR files are provided below.

- BT/ BTIP SIP Trunking North side:

Header Rule	Comment
outToBT	Modify user-agent header with IPBX/ESBC vendor version details before sending SIP messages to BT/BTIP

- Cisco CUCM South side:

Header Rule	Comment
outToPBXsipManip	Changes from and to header's uri-host to SBC's FQDN value and Modify user-agent header with IPBX/ESBC vendor version details before sending SIP messages to IPBX's

1.4.1 outToPBXsipManip

This manipulation is the core procedure for messages sent towards the PBX on access South side.

This manipulation performs the following operation:

- topology-hiding replacing From and To before delivering messages to a PBX. The actual replacing value is the local access IP of the SBC associated to the PBX for FROM and the PBX IP address for the TO header.
- Replacing UserAgent & Server header with ESBC & Cisco CUCM versions

Header Rule	Comment
outToPBXsipManip	CSBC # conf t CSBC (configure)# session-router sip-manipulation CSBC (sip-manipulation)# name outToPBXsipManip CSBC (sip-manipulation)# header-rules CSBC (sip-header-rules)# name my_To_hr CSBC (sip-header-rules)# header-name To CSBC (sip-header-rules)# action manipulate CSBC (sip-header-rules)# comparison-type case-sensitive CSBC (sip-header-rules)# msg-type request CSBC (sip-header-rules)# element-rules CSBC (sip-element-rules)# name My_To_er CSBC (sip-element-rules)# type uri-host CSBC (sip-element-rules)# action replace CSBC (sip-element-rules)# new-value *$REMOTE_IP* CSBC (sip-element-rules)# exit CSBC (sip-element-rules)# done CSBC (sip-header-rules)# name my_From_er CSBC (sip-header-rules)# header-name From CSBC (sip-header-rules)# action manipulate CSBC (sip-header-rules)# comparison-type case-sensitive CSBC (sip-header-rules)# msg-type request CSBC (sip-header-rules)# element-rules CSBC (sip-element-rules)# name My_From_er CSBC (sip-element-rules)# type uri-host CSBC (sip-element-rules)# action replace CSBC (sip-element-rules)# match-val-type ip CSBC (sip-element-rules)# new-value *$LOCAL_IP* CSBC (sip-element-rules)# exit CSBC (sip-header-rules)# name HR_CheckUserAgent CSBC (sip-header-rules)# header-name User-Agent CSBC (sip-header-rules)# action manipulate CSBC (sip-header-rules)# msg-type request CSBC (sip-header-rules)# methods INVITE CSBC (sip-header-rules)# new-value “ORACLE SBC/v.8.2.0. \\ CISCO_CUCM/v.12.0” sip-manipulation name outToPBXsipManip description Out to Cisco_CUCM split-headers join-headers header-rule name my_To_hr header-name To action manipulate comparison-type case-sensitive

```
            msg-type                        request
            methods
            match-value
            new-value
            element-rule
                name                            My_To_er
                parameter-name
                type                            uri-host
                action                          replace
                match-val-type                  any
                comparison-type                 case-sensitive
                match-value
                new-value                       $REMOTE_IP
        header-rule
            name                            my_From_er
            header-name                     From
            action                          manipulate
            comparison-type                 case-sensitive
            msg-type                        request
            methods
            match-value
            new-value
            element-rule
                name                            My_From_er
                parameter-name
                type                            uri-host
                action                          replace
                match-val-type                  ip
                comparison-type                 case-sensitive
                match-value
                new-value                       $LOCAL_IP
        header-rule
            name                            HR_CheckUserAgent
            header-name                     User-Agent
            action                          manipulate
            comparison-type                 case-sensitive
            msg-type                        request
            methods                         INVITE
            match-value
            new-value                       "ORACLE
SBC/v.8.2.0\\CISCO_CUCM/v.12.0"
        last-modified-by                admin@10.253.51.48
        last-modified-date              2019-04-18 12:12:04
```

1.4.2 outToBT

This manipulation is the core procedure for messages sent towards the BT/BTIP on Core North side and call in Core realm sip-interface out-manipulationid.

This manipulation performs the following operation :

- The manipulation performs replacing UserAGent

Header Rule	Comment
Header rule **HR_ChangeUserAgent**	CSBC # conf t CSBC (sip-manipulation)# name outToBT CSBC (sip-manipulation)# header-rules CSBC (sip-header-rules)# name HR_ChangeUserAgent CSBC (sip-header-rules)# header-name User-Agent CSBC (sip-header-rules)# action manipulate CSBC (sip-header-rules)# msg-type request CSBC (sip-header-rules)# methods INVITE CSBC (sip-header-rules)# new-value “ORACLE SBC/v.8.2.0. \\ CiscoCUCM/v.12.0” CSBC (sip-header-rules)# done CSBC (sip-header-rules)# exit sip-manipulation name outToBT description split-headers join-headers header-rule name HR_ChangeUserAgent header-name User-Agent action manipulate comparison-type case-sensitive msg-type request methods INVITE match-value new-value "ORACLE SBC/v.8.2.0\\CiscoCUCM/v.12.0" last-modified-by last-modified-date