Business Talk & Business Talk IP Configuration Guidelines with Patton SmartNode eSBC

version addressed in this guide: Patton SBC SmartNode V.3.20.4

Version of 05/04/2024

Information included in this document is dedicated to customer equipment (IPBX, TOIP ecosystems) connection to Business Talk & BTIP service: it shall not be used for other goals or in another context.

Table of contents

1. Goal of this document

The aim of this document is to provide configuration guidelines to ensure the interoperability between Patton Edge eSBC with Business Talk (BT) or Business Talk IP (BTIP) service from Orange Business Services, hereafter so-called “service”.

2. References documents

Title	Link
Trinity Release 3.14.X-Command Line Reference Guide	https://www.patton.com/manuals/trinity3.14cli.pdf
SmartNode SN500 User Manual	https://www.patton.com/manuals/50000093_SN500-UM.pdf
SmartNode SN5501 User Manual	https://www.patton.com/manuals/SN5501-UM.pdf

3. Prerequisites

3.1 Certificates

In case of encrypted SIP trunk architecture, mutual TLS configuration is mandatory in order to exchange public certificates with Orange BTalk infrastructure in both ways.
Customer public trusted certificates chain is used by both the eSBC to authenticate the connection with our infrastructure and Orange public trusted certificates chain is used by the eSBC to authenticate the connection

The customer must generate on the Patton eSBC a Certificate Singing Request (CSR) and request to a public Certificate Authority (CA) a public certificate.
Then only that the Root and intermediate Certificate Authorities (PEM format) must be communicated to Orange BTalk team.

3.2 Public DNS configuration:

Following requirements regarding Public DNS configuration must be follow :

- In eSBC configuration, public DNS is used for outgoing calls to PSTN (e.g. From iPBX/eSBC to BToI/BTIPoI)
- Internet-naming resolution (FQDN): either enter the IP addresses of 2 private DNS, that relay DNS queries to Internet, or enter the IPs of 2 accessible public DNS such as those of Orange (80.10.246.2, 80.10.246.129)

3.3 NTP

The configuration of NTP servers on the eSBC is not fully detailed (still some typical example is described in annex) in this document but it is mandatory to implement an NTP server (public reliable NTP server) on Patton Edge eSBC to ensure that the eSBC receives the current date and time. This is necessary for validating Certificates of remote parties during TLS “Handcheck”.

3.4 Firewall flows for BTIP over Internet and BT over Internet

Firewalls in the way of traffic between Patton Edge eSBC and Orange infrastructure have to be updated in order to open required ports for BT over Internet or BTIP over Internet vary concerning the UDP Media ports range.

For BTIP over Internet, please note the Orange infrastructure Media public IP termination is different from Orange infrastructure SIP Signaling public FQDN/Public IP termination.

Refer to the ‘Business Talk IP over Internet pre-requesites’ and “Business Talk STAS” documents provided by your sales/project manager team for more details about firewall rules needed to be open.

3.5 Orange BTalk/ BTIP specifications

The information in this chapter are the SIP trunk specifications required in order to interconnect Orange Business Talk network. The Enterprise SBC must be compliant with those specifications.
This information were used to define the configuration described in this document.

- ***Supported RFC's***
  - *RFC 3261 : Session initiation protocol*
  - *RFC 3264 : An offer/answer Model with the Session Description Protocol*
  - *RFC 3262 : Reliability of provisional responses in Session Initiation protocol (please refer to provisional response and PRACK section)*
  - *RFC 3311 : The Session Initiation Protocol UPDATE Method*
  - *RFC 3323 : A privacy Mechanism for the session Initiation Protocol*
  - *RFC 3325 : Session Initiation Protocol for Asserted Identity within Trusted Networks*
  - *RFC 3204 : MIME media types for ISUP and QSIG Objects*
  - *RFC 3550 : RTP : A transport Protocol for Real Time Applications*
  - *RFC 3711: SRTP: Secure Real-time Transport Protocol*
  - *RFC 3960 : Early Media and Ringing Tone generation in the Session Initiation Protocol*
  - *RFC 4566 : SDP: Session Description Protocol*
  - *RFC 4568: SDP: Security Descriptions for Media Streams*
  - *RFC 2833/4733 : RTP payload for DTMF digits, Telephony Tones and telephony signals*
  - *RFC 5806 : Diversion Indication in SIP*
  - *RFC 5009 : Private Header Extension to the Session Initiation Protocol for Authorization of early*

- ***Sip Methods supported:***
  - *INVITE*
  - *ACK*
  - *CANCEL*
  - *UPDATE (negotiated)*
  - *BYE*
  - *OPTIONS*

  Note : Sip methods not listed are not supported in this context

- ***SIP Message size specifications are:***
  - *SIP message limited to 4096 Bytes*
  - *SDP Body limited to 1024 Bytes*

- ***SIP signalling specifications are:***
  - *For **unencrypted architecture** we need to configure **UDP port 5060***
  - *For **encrypted architecture** (TLS) we need to configure **TCP port 5061***

- ***Media specifications are by default listed below and should be adapted to your Customer service offer:***
  - *For **unencrypted architecture** we need to configure RTP port **6 000 to 20 000***
  - *For **encrypted architecture** (TLS) we need to configuration **SRTP port 6 000 to 20 000 for Business Talk over Internet or SRTP port 6 000 to 38 000 for Business talk IP over Internet***

- ***Identification***
  - For Audit purpose eSBC **"User Agent"** connected to BTalk/BTIP infrastructure require following format: **"IPBX/UC Vendor < Product> <Version>.<build> \ Patton eSBC<SBC model> <Version>.<build>"**
  - Same requirement applies on Server Agent in provisional response

- ***Encryption specifications are:***
  - **TLS V.1.2**

The following Cipher list is supported as Cipher Client/Server:

- **TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384** (Recommended)
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA256

Mutual authentication activated.

✓ ***Codec/Packet Rate specifications are (prefer order list) :***

- G722 20 ms.(Only if specifically used)
- G.711 A-law 20 ms (or on demand specific G.711 µ--law 20 ms)
- G.729 20 ms (annexb = no)
- For BTIP over Internet and BTalk over Internet (TLS) only G.711 A-law 20 ms (or on demand specific G.711 µ--law 20 ms) is supported

✓ **Voice Activity Detection** (VAD) is **not supported**

✓ ***T.38 for FAX specifications are:***

- T.38 Fax over UDP
- T.38 payload size 20 ms or 40 ms
- NSF value 0
- Fax rate management method Transferred TCF
- UDP redundancy method T38UDPredundancy
- T.38 version parameter 0
- T.30 data V.21
- Data signaling rates: V.17 or V.29 or V.27ter
- Error Correction Method (ECM) Enabled
- Fax rate max 14400 bps
- SG3-G3 fallback method Either ANSam removal or CM removal
- Switching from voice mode to fax mode T.38 re-INVITE sent by called party

Note: For T.38, the Patton SmartNode SBC will be transparent. No adaptation will be done at SBC level as it requires DSP resources.

✓ ***DTMF transport specifications are:***

RFC 2833/4733

✓ ***Signalisation/ Media Tag specifications are:***

- ✓ DSCP 46 (EF)

✓ ***SIP Probing***

- BTalk/BTIP SIP Trunk relies on OPTIONS method to “probe” the eSBC, in dialog and out of dialog.
- The following answers are expected:
  - Out of dialog: 200 OK (or any error responses) if UE is up, nothing if down
  - In dialog: 200 OK if Call is active and 481 if Call is not active
- The UE could use OPTIONS with max-forward=0 to probe BTalk/BTIP SIP Trunk, in this case, Business Talk will send back a 200 OK.

✓ ***Call initiation***

- eSBC shall provide an SDP within his initial INVITE, delay offer (INVITE without SDP) is not supported.

- ***Media Session Modification/ Transfer – Call Forward:***
  - Modification of media (IP, codec, attributes ..) in reception/transmission based on UPDATE (With SDP) in Early Dialog and Re-INVITE in confirmed Dialog (with or without SDP)
  - Attributes “a=” must be equal to send only, recvonly, inactive, sendrecv.
  - In case of Call Forward, the diversion header must be provided by the UE.
  - Same Methods/Attributes/headers may be sent from BTalk/BTIP to UE.

- ***Ring back Tone and Early Media***
  - Presence of an SDP in provisional response does not indicate presence of a distant early media (only p-early-media indicate presence of distant early media).
  - On reception of a 180 (without SDP) from BTalk/ BTIP, eSBC must play local Ring Back Tone.
  - eSBC can indicate an early media, within presence of P-Early-Media header into his provisional response.

- ***Anonymous calls***
  - If anonymization is requested, the UE should:
    - Set privacy header to “user” with From containing Calling identity
    - Or: set privacy header to id with From containing anonymous (“anonymous” sip:anonymous@anonymous.invalid, P-A-I must contain the Calling party identification.
  - Same Settings could be used when Business Talk request anonymous calls.

- ***Number format specifications are:***
  - Called Number sent to Orange network must be at E164 format
  - Calling Number sent to Orange network must be in National format (0ZABPQMCDU or 00xxxxxxxx) or E164 format.

- ***Rerouting scenario:***
  - On reception of a Sip Error message, User Equipement must reroute in case of 408 et 50x (500/501/502/503/504/505/513)
  - Transmission of a SIP error message to BTalk/BTIP, UE must send 5xx if a rerouting is expected from BTalk/BTIP service.
  - It’s recommended to do not send 408 to BTalk/BTIP. If it’s the case, UE will be considered out of service until next Sip probing

- ***Call defection :***
  - 3xx Sip messages are not supported by BTalk/BTIP services. Those messages will be converted into SIP error messages.

4. Certified Architecture

4.1 Introduction to architecture components and features

This document describes “only” the main supported architectures either strictly used by our customers or 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…)

These configuration guidelines taken into account:

- Only considering Carrier North side of Patton Edge eSBC facing Business talk and BTIP offers.
- Consider the eSBC as this SIP North eSBC termination as a demarcation point for OBS, South eSBC side is out of Orange control and responsibility
- Stop considering the ecosystem behind the Patton Edge eSBCs on South Side (IPPBX vendor/version, mono vs multi vendors, complexity of the ecosystem,…)

Concerning the fax support, Business talk and BTIP support the following usage:

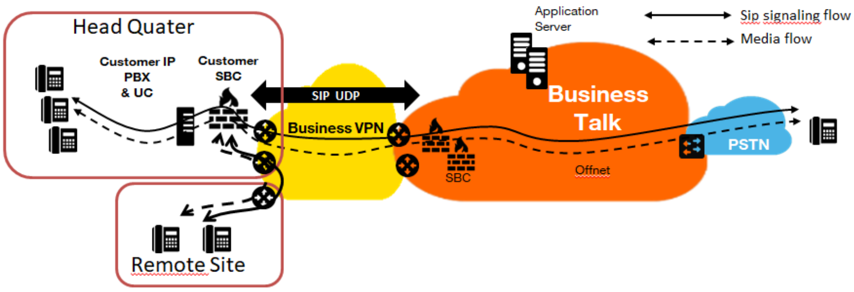
- fax servers connected to the IPBX* -and sharing same dial plan-, or as seperate ecosystems and separate dial plan.
- analog fax machines, usually connected behind and passing through Patton Edge eSBC
- Fax flows must handle via T.38 transport only.

Note: Fax communications via Business Talk will still be allowed but will no longer be officially supported by the Orange support teams from April 2023 for new customer implementations.

<u>* Please note</u> : This Patton Edge eSBC SIP North Carrier Side template configuration main objective is offering compliancy in front of BTIP / Btalk offers. Accordingly multi- vendor IPBX which added complexity must be addressed on Patton Edge eSBC SIP/T38 South side and are considered outside of OBS responsibilities.

4.2 Architecture with Patton “customer” SBC with OBS SIP North Carrier configuration

4.2.1 Unencrypted SIP Trunk (UDP)

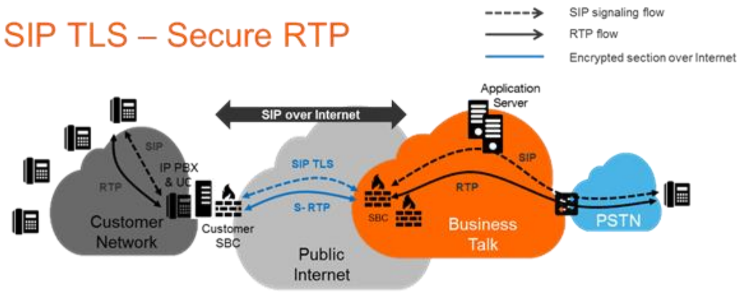


In this architecture:

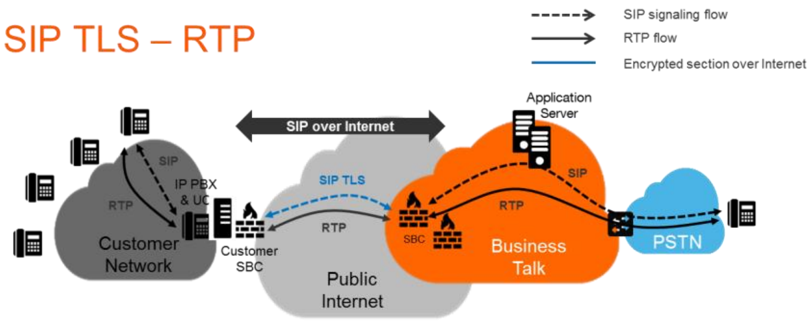
- Both ‘SIP trunking’ and RTP media flows between endpoints and the Business Talk/BTIP are anchored by the “customer SBC”:
- For Head Quarter & remote sites, media flows are routed through the Customer SBC and the main BVPN connection.

4.2.2 Encrypted SIP Trunk Over Internet (TLS)

- SIP TLS + Secured RTP: all SIP messages and media packets are encrypted on the public internet between Orange and the customer Internet SIP & Media endpoints. This is the level of encryption recommended by default by Orange to ensure security & privacy



- SIP TLS + (unencrypted) RTP: all SIP messages are encrypted on the public internet between Orange and the customer internet SIP endpoints. RTP flows are shared without encryption between the customer media endpoints and Orange backbone. This solution is less recommended by Orange, but allowed as customers can have encryption/decryption limitations



4.3 Parameters to be provided by customers to access the service.

Unencrypted SIP Trunk through BVPN

Depending on Customer architecture scenario selected, several IP addresses (V4) have to be provided by the Customer. The table below sum-up the IP Address (marked in red) required according to the scenario.

Applicable to all Session Border Controller with BTIP or BTalk over BVPN

Customer SBC – architecture with eSBC	Level of Service	@IP used by service	
1 Single Customer SBC	No redundancy	eSBC @IP	
2 Customer 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	eSBC1 @IP	eSBC2 @IP
2 Customer SBC in Load Sharing	- **Local redundancy**: both SBC are hosted on the same site OR - **Geographical redundancy** both SBC are hosted on 2 different sites	eSBC1 @IP eSBC2 @IP	

Encrypted SIP Trunk through Internet

Applicable to Customer SBC with BTalk over internet only (International)

Customer SBC – architecture with eSBC	Level of Service	@IP used by service	
1 Single Customer SBC	No redundancy	eSBC1 @IP or Public FQDN	
2 Customer 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	eSBC1 @IP or Public FQDN	eSBC2 @IP or Public FQDN
2 Customer SBC in Load Sharing	- **Local redundancy**: both SBC are hosted on the same site OR - **Geographical redundancy** both SBC are hosted on 2 different sites	eSBC1 @IP or Public FQDN eSBC2 @IP or Public FQDN	

Applicable to Customer SBC with BTalk IP over internet only (French)

Customer SBC – architecture with eSBC	Level of Service	@IP used by service	
1 Single Customer SBC	No redundancy	eSBC1 FQDN Type A	
2 Customer SBC Nominal / Backup mode (**DNS Resiliency model**)	- **Local redundancy**: both SBC are hosted on the same site OR - **Geographical redundancy** both SBC are hosted on 2 different sites	eSBC public FQDN DNS Type **SRV**	
2 Customer SBC Nominal / Backup mode (**SIP Resiliency model**)	- **Local redundancy**: both SBC are hosted on the same site OR - **Geographical redundancy** both SBC are hosted on 2 different sites	eSBC1 FQDN Type A *	eSBC2 FQDN Type A*
2 Customer SBC in Load Sharing (**SIP Resiliency model**)	- **Local redundancy**: both SBC are hosted on the same site OR - **Geographical redundancy** both SBC are hosted on 2 different sites	eSBC1 FQDN Type A* eSBC2 FQDN Type A*	

* Only eSBC public FQDN's SIP Termination will be supported, eSBC public IP's Termination will not.

4.3.1 Information and Syntax

The naming of the different objects created (network interface, rules names, routing or mapping tables' names …) must be respected in order to guarantee the coherence of the configuration and make the configuration check by OBS easier in case of issue.

Few parameters highlighted in "Green" color (IP Address, capacity, …) in this document are given as example and must be replaced by the real specific value of the corresponding interconnection.

Several tables in the following Chapters, will contain lines in "**Grey**" color. Those lines are indicated as example and reminder of the existing configuration of the "south" side (IPPBX side) inside the SBC. If the SBC used is a new one without existing configuration, you must replace those "**Grey**" lines according to the specifications of your IPPBX/UC environment you want to interconnect to BTalk/BTIP network through the eSBC.

Example

Description	Host/domain	Server Lookup	Port	Protocol
Orange_BTalk	Nominal: <BT_Nominal_IP> or <BT_Nominal_FQDN> Backup: <BT_backup_IP> or BT_backup_FQDN	IP/FQDN	5060 or 5061	SIP/UDP or SIP/TLS
Patton SBC or IPBX	*Nominal: 6.6.77.10* *Backup: 6.6.77.11*	IP/FQDN	*5060*	*SIP/UDP*

4.4 Business Talk & BTIP Patton SmartNode eSBC certified versions

Patton SmartNode eSBC – software versions				
Reference product	Hardware or Virtual Model	Software Major version	Certified "Loads"	Certification
SmartNode SN500 (SBC)	Hardware: SN500/4B/EUI (soft-DSP, no HW DSP**)	v. 3.20	Load(s) 3.20.4 (*)	✓BTIP, BTalk, BTIP over Internet, BTalk over Internet
SN5300 (SBC)	Hardware: SN5301/4B/EUI SN5301/4B2G/EUI SN5301/4B4G/EUI (soft-DSP, no HW DSP**)	v. 3.20	Load(s) 3.20.4 (*)	✓BTIP, BTalk, BTIP over Internet, BTalk over Internet
SN5500 (SBC)	Hardware: SN5501/8B/EUI SN5501/16B/EUI	v. 3.20	Load(s) 3.20.4 (*)	✓BTIP, BTalk, BTIP over Internet, BTalk over Internet
SN5530 (SBC+ BRI GW)	Hardware: SN5531/2BIS4VHP/EUI SN5531/2BIS4VHP4G/EUI SN5531/2BIS4VHPAVA/EUI SN5531/2BIS4VHPAVB/EUI SN5531/2BIS4VRHP/EUI SN5531/4BIS8VHP/EUI SN5531/4BIS8VHPAVA/EUI SN5531/4BIS8VHPAVB/EUI SN5531/4BIS8VRHP/EUI SN5531/8BIS16VHP/EUI SN5531/8BIS16VHP2G/EUI SN5531/8BIS16VHP4G/EUI SN5531/8BIS16VHPAVA/EUI SN5531/8BIS16VHPAVB/EUI SN5531/8BIS16VRHPAVA/EUI SN5531/8BIS16VRHPAVB/EUI	v. 3.20	Load(s) 3.20.4 (*)	✓BTIP, BTalk, BTIP over Internet, BTalk over Internet
SN5541 (SBC+ FXS GW)	SN5541/2JS2V/EUI SN5541/4JS4V/EUI SN5541/8JS8V/EUI SN5541/2JS2VAVA/EUI SN5541/2JS2VAVB/EUI SN5541/4JS4VAVA/EUI SN5541/4JS4VAVB/EUI SN5541/8JS8VAVA/EUI SN5541/8JS8VAVB/EUI SN5541/2LL2V/EUI SN5541/4LL4V/EUI SN5541/2JS2JO4V/EUI SN5541/4JS4JO8V/EUI	v. 3.20	Load(s) 3.20.4 (*)	✓BTIP, BTalk, BTIP over Internet, BTalk over Internet
SN5551 (SBC+ FXS/BRI GW)	Hardware: SN5551/2BIS2JS4VHP/EUI SN5551/2BIS2JS4VHPAVA/EUI SN5551/2BIS2JS4VHPAVB/EUI SN5551/2BIS4JS8VHPAVA/EUI SN5551/2BIS4JS8VHPAVB/EUI SN5551/4BIS2JS8VHPAVA/EUI SN5551/4BIS2JS8VHPAVB/EUI SN5551/4BIS4JS8VHP/EUI SN5551/4BIS4JS8VHPAVA/EUI SN5551/4BIS4JS8VHPAVB/EUI	v. 3.20	Load(s) 3.20.4 (*)	✓BTIP, BTalk, BTIP over Internet, BTalk over Internet

SN5571 (SBC + PRI GW)	Hardware: SN5571/1E15V30HP/EUI SN5571/1E15V30HPAVA/EUI SN5571/1E15V30HPAVB/EUI SN5571/1E15VHP/EUI SN5571/1E30VHP/EUI SN5571/1E30VHPAVA/EUI SN5571/1E30VHPAVB/EUI SN5571/2E30VRHP/EUI SN5571/2E30VRHPAVA/EUI SN5571/2E30VRHPAVB/EUI	v. 3.20	Load(s) 3.20.4 (*)	✓BTIP, BTalk, BTIP over Internet, BTalk over Internet
SN5600 (SBC) Appliance Server	Hardware: SN5600/4B/EUI (soft-DSP, no HW DSP**)	v. 3.20	Load(s) 3.20.4 (*)	✓BTIP, BTalk, BTIP over Internet, BTalk over Internet
Virtual SmartNode	Software / virtual model Catalog # CBFL-VSN-SBC (soft-DSP, no HW DSP**) Cloud Service Plan: CSP-C2E/STANDARD min required	v. 3.20	Load(s) 3.20.4 (*)	✓BTIP, BTalk, BTIP over Internet, BTalk over Internet

* Minimum Load for implementation, last most up-to-date load is recommended by Patton.
** Without transcoding capability (but with codec negotiation),no local RBT generation capability is generated by the Patton eSBC

Note:

Patton SBC technical documentation is available on the Web:
https://www.patton.com/products/voip-comparison.asp
-> click on the tab “eSBC”
-> in the listed table, click on the product column: for example SN500, SN53xx, SN55xx, vSN …
-> under the selected product, click on “Related Information”

5. Configuration Guidelines

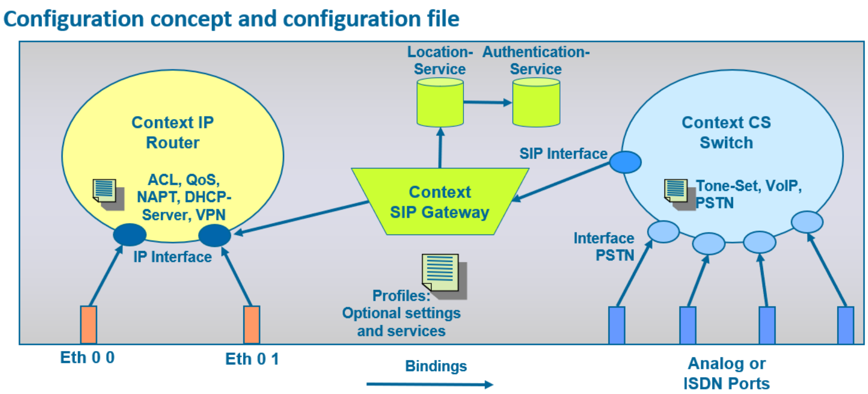
5.1 Patton Global configuration

This chapter describes the general configuration concept of Patton eSBC and the global configuration parts, before we explain the specific configuration for OBS Business Talk & BTIP SIP-Trunk in the next dedicated chapters.

We cover all global and system settings to be performed on Patton eSBC as well as all SIP and Media settings that are common for both VoIP Access Profiles of OBS (OBS Business Talk & BTIP Carrier North unencrypted SIP configuration for Patton eSBC (UDP) and OBS Business Talk over Internet & BTIP over Internet Carrier North encrypted SIP configuration for Patton SBC (TLS) OBS Business Talk over Internet & BTIP over Internet Carrier North encrypted SIP configuration for Patton SBC (TLS)).

The next chapters 3.2 and 3.3, dedicated to SIP/UDP and SIP/TLS profiles, then cover the specific mandatory settings required by those two access profiles.

5.1.1 Objects



This chapter describes the Patton SBC necessary configuration steps for a correct interoperability with the Orange Business Trunking Business Talk.

Patton eSBC's configuration concept is based on the principle shown on the figure above.

The physical interfaces are called Ethernet ports. Each of them (Eth 0 0 and Eth 0 1) is bound through the configuration to one or more (logical) IP Interfaces, which have to be previously defined in the Context IP Router.

Then the Context SIP Gateway builds the gateway between IP and the internal Call Routing (Context CS Switch). Interfaces used in the Call Routing are SIP and TDM (Analog / ISDN) interfaces. In a pure (IP-IP) SBC scenario, we only use SIP interfaces in the call router of the SmartNode.

In general, the physical interface Eth 0 0 is bound to the IP interface WAN (public side) and the physical interface Eth 0 1 to the IP interface LAN, whereas it is just a convention. SIP interfaces from the Call Router are bound to the SIP Gateway, which in turn is bound to the IP Interfaces (Network Interfaces) from the Context IP.

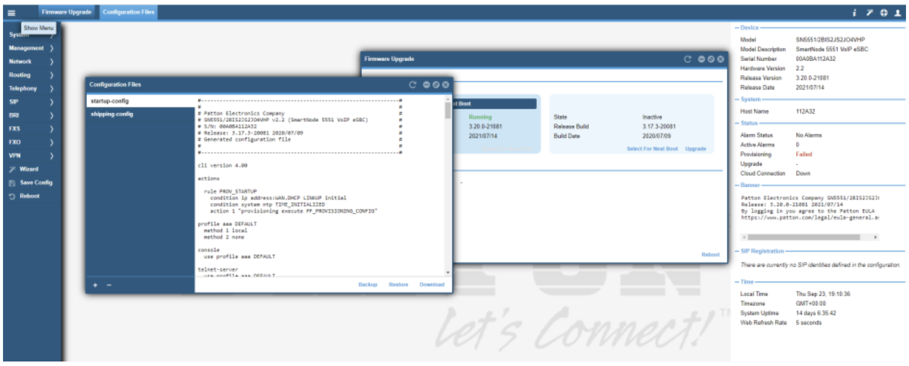
It is highly recommended to have a dedicated Interface for SIP Trunking Service provider like OBS BTIP / BTalk in order to differentiate the internal (private) from the external (public) SIP traffic.

Main configuration parts on Patton eSBC:

- Set a superuser account (see Annex Set a superuser account)
- DNS, NTP (see Annexes DNS Server configuration, NTP server configuration)
- IP Configuration (addresses, static routing)
- Physical port bindings
- Media/VoIP profiles
- SIP Location Service
- TLS Configuration
- SIP Gateway / SIP Trunk
- Call routing / Number & SIP URI manipulation
- SIP Interfaces / Header manipulation
- Hunt groups / failover configuration
- General SIP settings

Notes:

- *All SmartNode models, except the entry level model SN500, have an integrated Web UI available for the administration of the unit.*
- *Most of the configuration elements listed in this document are available through the Web UI, but some of them can only be configured through CLI (guidelines flagged "Only via CLI" in the next chapters) through telnet / SSH connection.*



Patton Web User interface

Note: Screenshot taken on a Patton SN5551

Warning:
Before applying the configuration described in this document, it is strongly recommended to proceed to a Backup of your Patton eSBC configuration through the CLI command **copy running-config config:<backup_cfg_name>** then save the configuration file on your laptop. When you have finished the configuration, proceed to a "save config" of your SBC configuration through the CLI command **copy running-config startup-config** and do again of Backup of your new configuration.

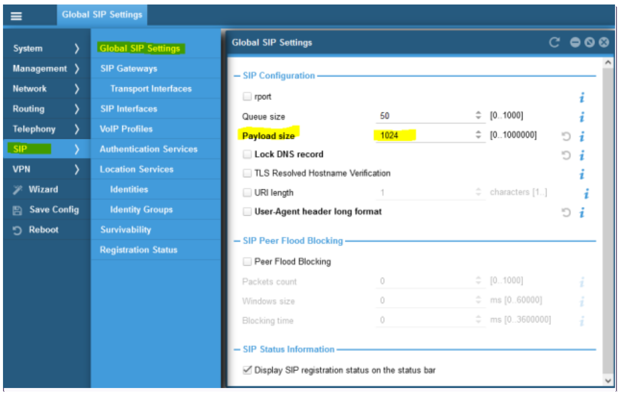
Remark :

For more information regarding backup and restore process on Patton eSBC devices, please consult the CLI Reference Guide, Chapter "Configuration File Handling" (current version on the date of editing this guide:
https://www.patton.com/manuals/Trinity3.14cli.pdf , Chapter 7 "Configuration File Handling")

5.1.2 SDP Body size limitation

Orange BTalk/BTIP specifications require to limit the size of the SIP message to 4096 Bytes and SDP Body to 1024 Bytes.

The maximum size of the SIP SDP body can be configured under the global SIP settings of the eSBC. Modify the default value of 4000 bytes to the specified maximum SDP size of 1024 bytes.

Actions	Screenshot
Configure max. SDP body size	Via Web UI: Open the menu SIP > Global SIP Settings, and set the Payload size to 1024 (bytes).  Via CLI: `sip` `  max-payload-size 1024`

Commenté [CSO1]: This is redundant with § 2.5.2 & 2.6.5
Commenté [BR2R1]: We keep it here (as it is global) and I removed it from 2.5 and 2.6
Commenté [CSO3R1]: OK agreed

5.1.3 Configure Media System Port range

The RTP Port Range configuration allows to define the total amount of RTP ports to be used by the eSBC on global level. This means that the RTP Port Range must take all SIP interfaces into consideration, not only the ones facing the SIP-Trunk to OBS BTIP/BTalk.

Port Pairs Considerations:
For Patton SmartNode eSBC: The number of RTP Port Pairs must be configured slightly larger than the actual number of ports required to support the projected number of calls. We recommend you over-allocate the number of port pairs by approximately 25 - 30% above the number of calls you want to support. In all call scenarios bellow, the eSBC may temporarily need more RTP ports, so this overallocation is useful for such specific use cases:

- Redirect services
- Supplemantary Services
- Call Forwarding
- Failover to Fax

SBC Reserved Ports - Example

Projected number of calls	Approximate number of Port pairs	Applies To
20 sessions	120 (Shared for all SIP Trunk)	Audio calls only *

* Multiple audio and video stream proxy calls will require twice the number of RTP port pairs with the examples provided above.

To determine the last corresponding port number:

SN500 Example: Given starting port number 6000 and the number for port pairs is 60 (Shared for all SIP Trunk). There are 60 pairs, meaning there are 120 individual ports. 6000 + (120-1) = 6119
SN500 Example: in case of SIP-Trunk connectivity to OBS with following inputs:

- 20 allocated parallel voice calls possible through the SN500 taken 25% of burst,
- Start port of the range: 6000

you will have to allocate:

- 2 x 30 = 60 RTP ports for the OBS SIP-Trunk, because each VoIP call requires 2 RTP ports
- Additional 2 x 30 = 60 RTP ports for the IPPBX / LAN side (2 RTP ports / call OBS – IPPBX)
- Total: 60 + 60 = 120 RTP ports

CLI Command	Parameter	Value
rtp-port-range	<portRangeLow>	6000
	<portRangeHigh>	6119

Actions	Screenshot
Configure RTP Port Range	Only via CLI `rtp-port-range 6000 6119`

5.1.4 User-Agent and Server header format

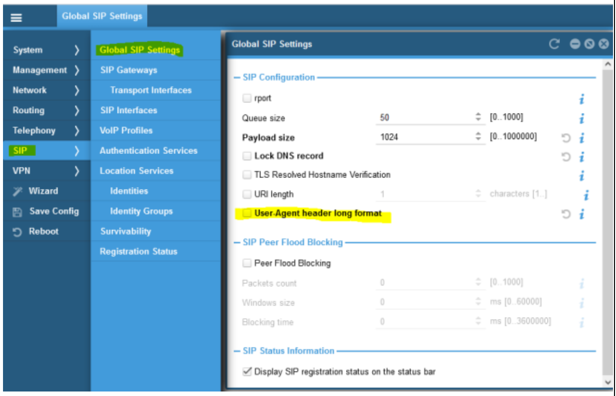
There is a possibility to use either a long (default) or a short User-Agent and Server header format, sent by Patton eSBC.

Example of the long format: "`Patton SN500/4B 00A0BA10DD86 3.20.2-21122 1.3 M5T SIP Stack/4.2.28.153`"

Example of the short format: "`Patton SN500 00A0BA10DD86 3.20.2-21122`"

We suggest to change to the short format because it will be concatenated with the header from the IPPBX behind the SBC, as required by OBS, which makes the whole header content much longer than usual.

Only the mentioned configuration element below (User Agent header) has to be modified and has effect on both UA and Server headers sent by the eSBC.

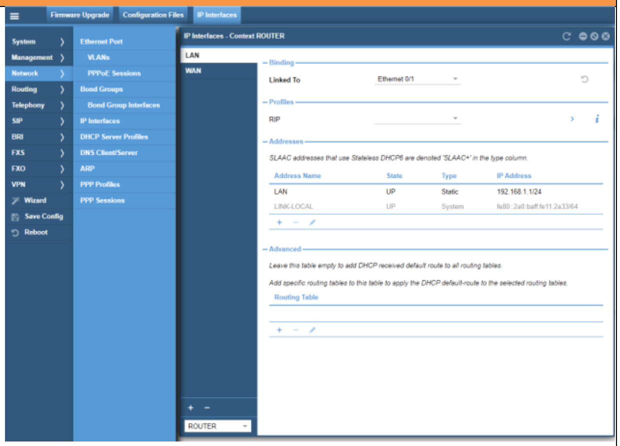
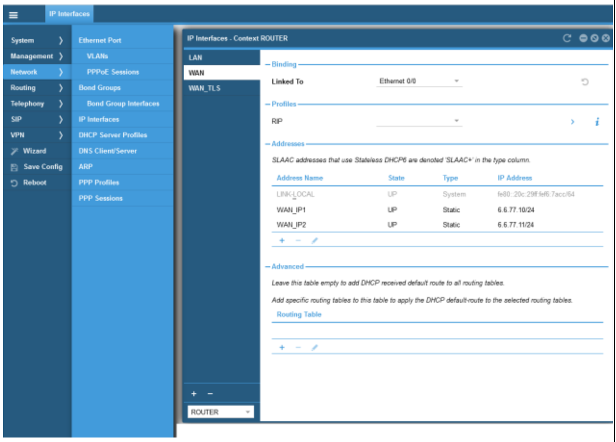
Actions	Screenshot
1. Change the User-Agent and Server header formats	Via Web UI: Open the menu SIP > Global SIP Settings, and **uncheck** the box "User-Agent header long format".  Via CLI: `sip` `  user-agent-header-format short`

2. Apply header manipulation	Apply the header manipulation as explained in the chapter OBS-specific User-Agent and Server headers, in order to send the specific content in User-Agent and Server headers, as specified by OBS (<IPBX Vendor v.X.X + SBC vendorV.X.X>)

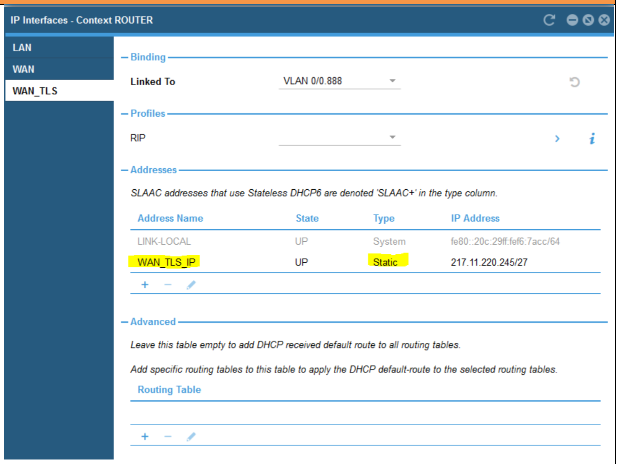
5.1.5 Configure Network Interfaces (Context IP)

The Network > IP Interfaces menu path allows you to configure the IP addresses (both IPv4 and IPv6) for the Ethernet ports and VLANs.

Actions	Screenshot
Create IP Interfaces and define their IP addresses (please consider the warning at the end of this table)	Web UI: Open the menu Network / IP Interfaces. • Create the IP interfaces LAN • For the scenario SIP/UDP, we suggest a convention which consists in creating a WAN network interface with the name WAN, and for the SIP/TLS scenario a WAN interface with the name WAN_TLS. Note that it is possible to create both, or even several network interfaces, which can then be bound to the physical Eth interfaces through VLANs. If you intend to deploy Patton SmartNode eSBC only for one of the two scenarios (UDP or TLS), consider only the related interface and address creation below. • Create static IP addresses using “+” button under address. Use following convention for the public / trunk side: o SIP/UDP: under the created WAN interface enter IP address names WAN_IP1 (local main) and optionally WAN_IP2 (local backup) that will be used for local IP redundancy. o SIP/TLS: we don’t implement local IP redundancy for TLS. Under the created WAN_TLS interface enter only one IP address name: WAN_TLS_IP. The following screenshots show IP address names with the assigned static IP address examples.

Actions	Screenshot
	 

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>



Do not consider the part ‘Binding’ at the moment. This one will be automatically set later on, once you proceed to the binding from the physical interface configuration (see Configure physical interfaces).

Via CLI:

```
context ip ROUTER

  interface WAN
    ipaddress WAN_IP1 <IP_address1>/<netmask>
    ipaddress WAN_IP2 <IP_address2>/<netmask>

  interface WAN_TLS
    ipaddress WAN_TLS_IP <IP_address3>/<netmask>
  interface LAN
    ipaddress LAN_IP <IP_address>/<netmask>
```

</td></tr>
</table>

Warning:
Please be careful when you change IP settings on the unit. If you set a wrong IP address for the interface you are using for the device administration, you will lose the management IP connectivity to the SN immediately. In such a case you have to reconnect to the newly assigned IP address or use the console connection (not available on all eSBC models).

5.1.6 DSCP profile

The DCSP service-policy profile must be created before network interface configuration, because it will have to be applied to the WAN interface once it is created.

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td>Set Signaling and Media DSCP Tag

(the traffic class “local voice” will be applied to RTP DSCP and the traffic class local-signaling will be applied to SIP DSCP)</td><td>Only via CLI:


```
profile service-policy SP_WAN_OUT
```

</td></tr>
</table>

Actions	Screenshot
	`source traffic-class local-voice` `  set ip dscp 46` `source traffic-class local-signaling` `  set ip dscp 46`

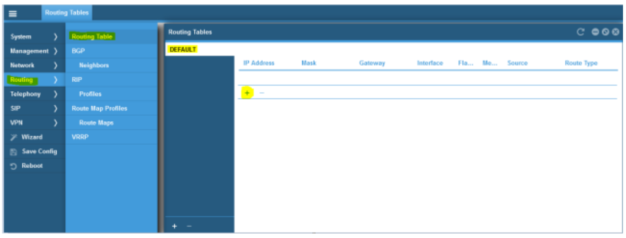
Note: this traffic class will be applied to the WAN interface in the next step (for the media DSCP tagging part) and to the context SIP Gateway (for the signaling DSCP tagging part).

5.1.7 Apply DSCP profile

Actions	Screenshot
Apply the DSCP profile to the network interface WAN (for Media tagging)	Only via CLI: Apply the previously created service policy SP_WAN_OUT to the interface WAN and/or WAN_TLS in outgoing direction. `context ip ROUTER` `  interface WAN` `    use profile service-policy out SP_WAN_OUT` `context ip ROUTER` `  interface WAN_TLS` `    use profile service-policy out SP_WAN_OUT`

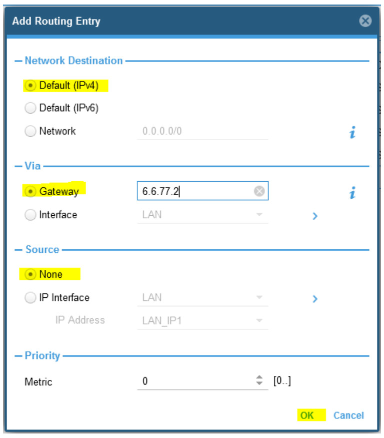
5.1.8 Configure Static Routes

The *Routing* > *Routing table* menu path and CLI *context ip ROUTER* > *routing-table DEFAULT* allows the administrator to manually specify the next hop routers used to reach other networks. This is also where you specify the default routes for the connected IP networks (which use 0.0.0.0/0 as destination and mask).

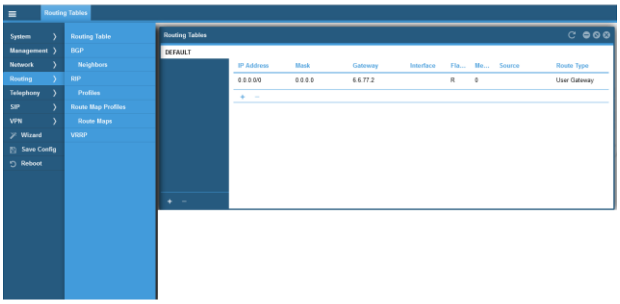
Actions	Screenshot
Adding a static default route	Web UI: Open the menu Routing / Routing Table, and under DEFAULT use the "+" button to create a static IP route to the DEFAULT routing table. 

In the ‘Add Routing Entry’ window, select following options:

- Network Destination: select ‘Default (IPv4)’ (or ‘Default (IPv6)’ in case of IPv6’
- Via: select Gateway and enter the IP address of the default gateway; Optionally select Interface and select a configured IP Interface if this interface should be used as default route.
- Source: select ‘None’ -> means default route valid for ingress IP traffic from all configured interfaces; optionally select Interface and chose a configured IP interface if the defined route should be applied to ingress IP traffic from that particular interface only.



In the example below, we created a static default route via gateway with IP address 6.6.77.2.



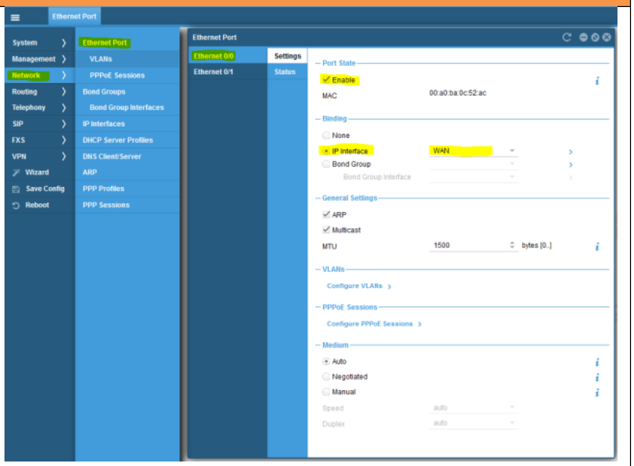
Remarks:

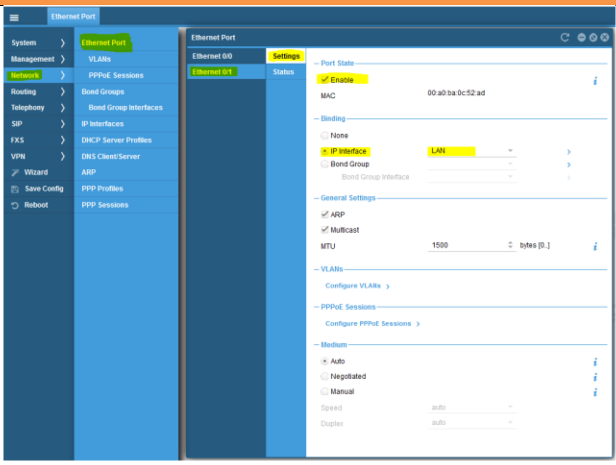
	The created static route will by default be applied to the corresponding interfaces subnets. In the provided example, let's assume the WAN interface has the IP address 6.6.77.10/24, so the default router 6.6.77.2 will be applied to the WAN interface subnet 6.6.77.0/24. Via CLI: `context ip ROUTER` `  routing-table DEFAULT` `    route 0.0.0.0/0 gateway <def_router_ip_addr> metric 0` Tip: You can verify your IP routing configuration with the following CLI command, which provides the effective status of the IP route table : `show route`

5.1.9 Configure physical interfaces

Once the IP Interfaces and IP address have been created, the physical interfaces (i.e. Ethernet ports) can be bound to them. The commonly used convention is to assign Eth 0 0 to the WAN interface (public network towards OBS SIP-Trunk) and Eth 0 1 to the LAN interface (private network towards the IPPBX), but this convention is not mandatory, so you are free to choose your preferred assignment.

Actions	Screenshot
Bind Eth 0 0 physical interface to WAN interface and enable it	Web UI: Open the menu Network / Ethernet Port / Settings, then: • in the configuration item “Binding” select the WAN or WAN_TLS interface (depending on the planned deployment) in the drop-down list with the IP interface names, that have been created in the previous step. • enable the port by checking the box “Enable” under the item “Port State”

Actions	Screenshot
	 Via CLI: SIP/UDP deployment: `port ethernet 0 0` `  bind interface ROUTER WAN` `  no shutdown` SIP/TLS deployment: `port ethernet 0 0` `  bind interface ROUTER WAN_TLS` `  no shutdown`
Bind Eth 0 1 physical interface to LAN interface and enable it	Web UI: Open the menu Network / Ethernet Port / Settings, then: • in the configuration item “Binding” select the LAN interface in the drop-down list with the IP interface names, that have been created in the previous step • enable the port by checking the box “Enable” under the item “Port State”

Actions	Screenshot
	 Via CLI: `port ethernet 0 1` `  bind interface ROUTER LAN` `  no shutdown`

In case VLAN’s have to be used, these have to be configured under Network / Ethernet Port / VLANs and assigned to the corresponding IP interface in exactly the same way as explained above.

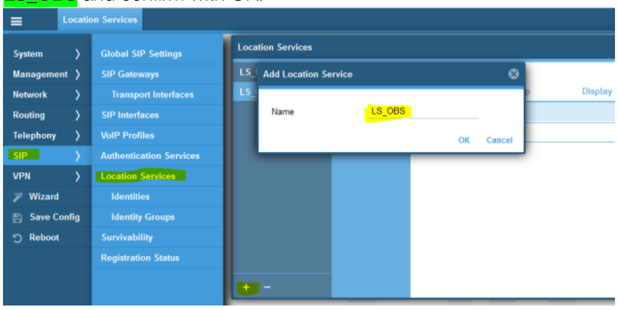
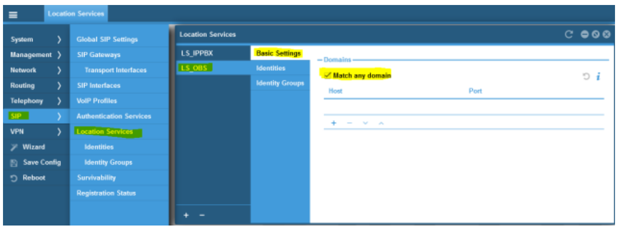
If required, speed and duplex settings can be modified under Medium. The default settings is Auto (auto-negotiation).

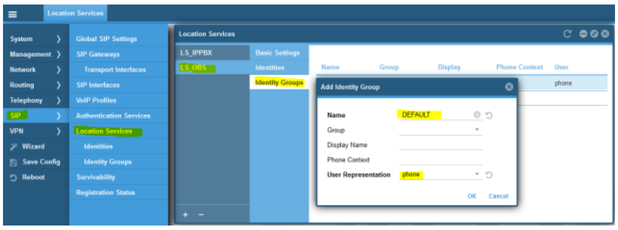
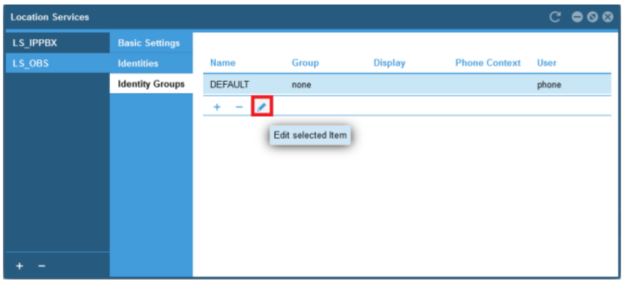
5.2 OBS Business Talk & BTIP Carrier North unencrypted SIP configuration for Patton eSBC (UDP)

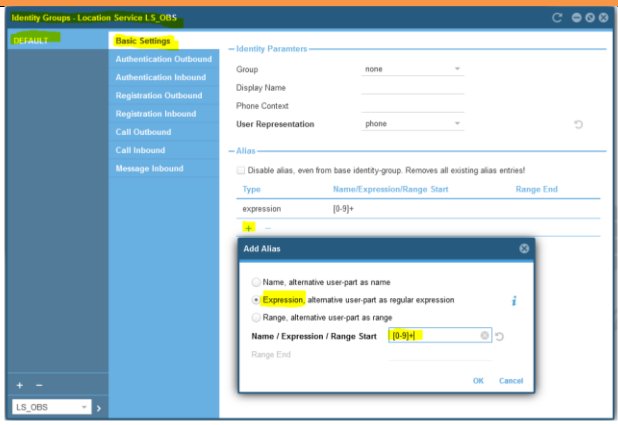
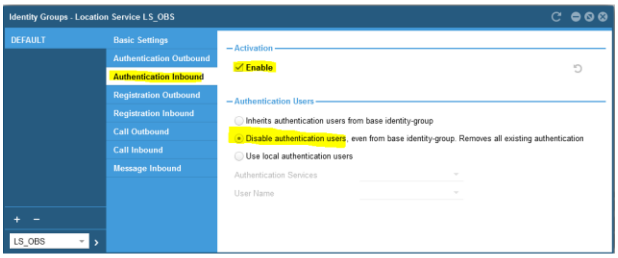
5.2.1 Configure Location Service

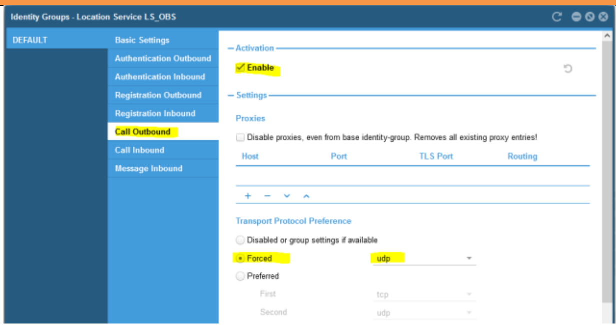
The Location Service on Patton eSBC is used to define specific incoming or outgoing authentication credentials (if required), registration parameters or to additionally restrict incoming SIP requests to only certain domain names or SIP URI's (through regular expressions).

In our case we use the Location Service only to add 'user=phone' to the Request URI, From, To, PAI headers, especially in order to specify that the user-part of the URI should be interpreted as a telephone number (tel-URI).

Actions	Screenshot
Create Location Service LS_OBS	Via Web UI: Open the menu SIP > Location Services then click on '+' to create a new Location Service, enter the name LS_OBS and confirm with OK. 
Allow « match any domain »	Select the newly created Location Service LS_OBS and under Basic Settings check the box "Match any domain", which means that any host part of incoming SIP URI's is accepted (not filtered). Menu: SIP > Location Services > LS_OBS > Basic Settings 

Actions	Screenshot
	Note: to restrict incoming SIP requests only to a certain host part of incoming SIP URI's, create only the corresponding host(s) and uncheck the box "Match any domain".
Create the Identity Group DEFAULT	Menu: SIP > Locations Services > LS_OBS > Identity Group Name : enter 'DEFAULT' User Representation : select 'phone' 
Modify additional parameters of the Identity Group DEFAULT	Open the created Identity Group DEFAULT by clicking on the pencil (edit) button:  Under Alias settings click on '+' and select 'Expression' to configure possible contents of user part:

Actions	Screenshot
	 Enter '[0-9]+' into the field Expression, which means that in case of outgoing SIP requests any digit combination in From header will be handled. If you want to restrict outgoing SIP requests to only a certain number range in the user part of From header (for example exclusively to the assigned DDI range for this SIP-trunk), then select Range and insert the number range into this field.
Disable inbound authentication	Under 'Authentication Inbound' select the checkbox 'Enable' under 'Activation', then under 'Authentication users' select the option 'Disable authentication users…'. This means that incoming SIP requests that passed the previous checks (domain, user-part) will not additionally be authenticated / challenged. 
Set UDP protocol for outgoing calls	Under 'Call Outbound' select the check-box under 'Activation' and in under 'Transport Protocol Preference' select the option 'Forced' and select 'udp'.

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>

Note: there is also an option to set a preference order (ex: 1) UDP, 2) TCP) but we explicitly use only UDP because it is specified so for the SIP-Trunk for BTIP/BTalk without encryption.</td></tr>
<tr><td></td><td>Via CLI:

location-service LS_OBS
  match-any-domain

  identity-group DEFAULT

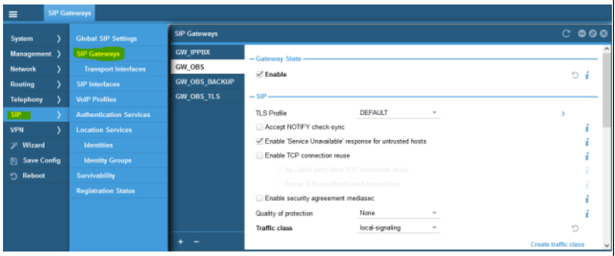
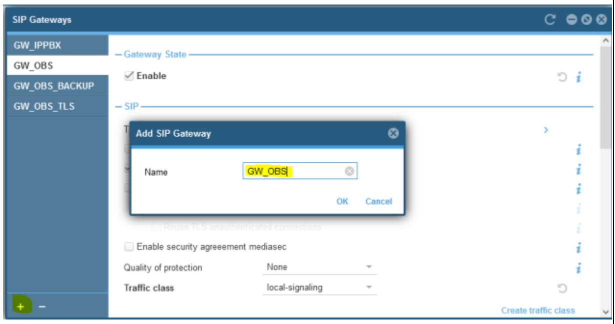
    alias expression [0-9]+
    user phone

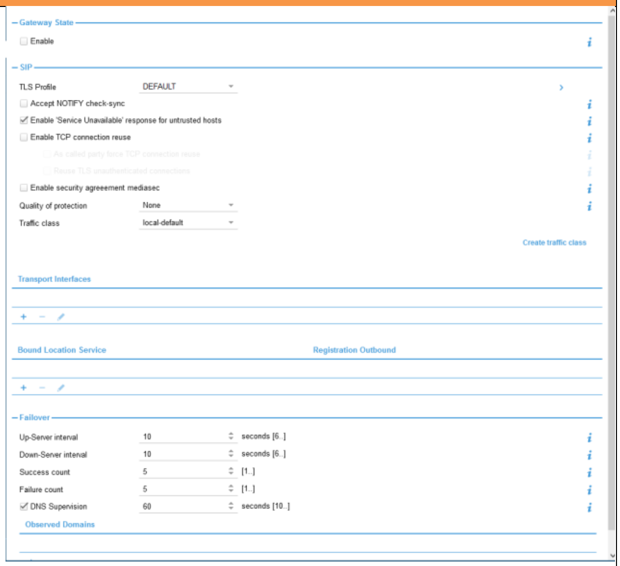
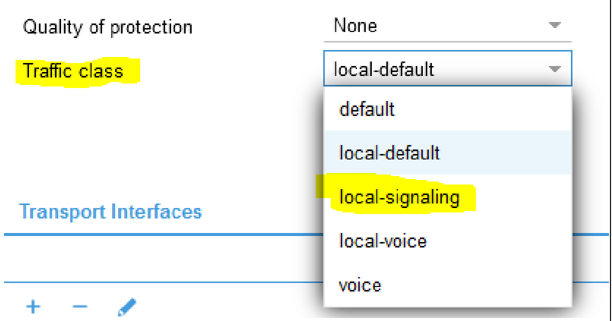
    authentication inbound
      authenticate none

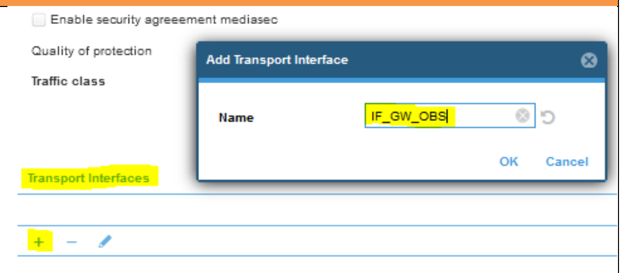
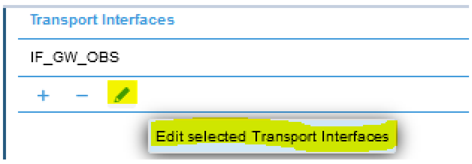
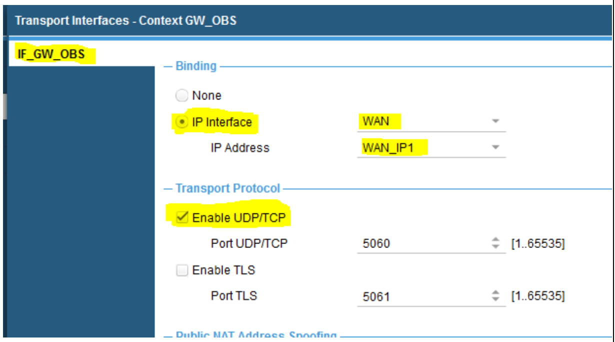
    call outbound
      transport-protocol force udp</td></tr>
</table>

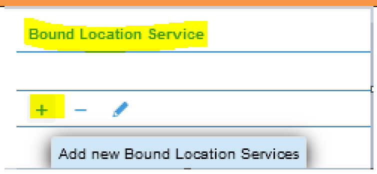
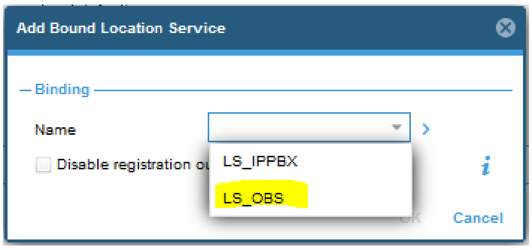
5.2.2 Configure SIP Gateway

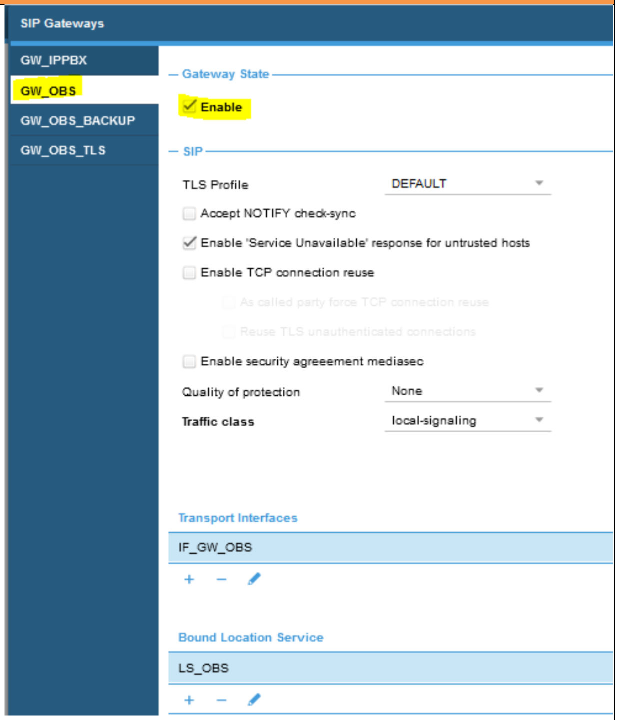
SIP-Gateway / main local IP-address

Actions	Screenshot
Access the SIP Gateway menu	Via Web UI: Open the **menu** SIP > SIP Gateways 
Create the SIP Gateway GW_OBS	Click on '+' at the bottom left to create a new SIP Gateway, enter the name 'GW_OBS' and confirm with OK.  The figure below shows how the newly created SIP Gateway looks like by default before the necessary changes are made:

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>
</td></tr>
<tr><td>Select the correct traffic class for DSCP tagging for SIP signaling</td><td>Select the traffic class 'local-signaling'. Important: this traffic class has been configured in the DSCP profile (profile service-policy SP_WAN_OUT), as explained under Global configuration in the chapter DSCP profile. Select this setting in order to ensure the corresponding packet tagging of outgoing SIP messages towards the BTIP/BTalk SIP-Trunk. Leaving the default setting here would mean no DSCP tagging, so don't miss this part.

</td></tr>
<tr><td>Create transport interface</td><td>As any other name of variable (by convention in capital letters), you are also free to define a name for the transport interface inside the SIP Gateway. It is through this interface that the binding with an IP address from the context IP / Interface is done. By convention we set the name IF_GW_OBS:</td></tr>
</table>

Actions	Screenshot
	
Edit the the created transport interface	Edit the settings of the newly created transport interface: 
Modify the default settings inside the transport interface	Under binding select 'IP interface' and select the existing IP interface 'WAN' and select the IP address 'WAN_IP1', both created previously in the the Network Interface Configuration (Context IP). Under 'Transport Protocol' check the box 'Enable UDP/TCP'. Leave the default port setting to 5060. Do NOT enable TLS, leave it on the default disabled setting. 
Bind the location service LS_OBS	Under 'Bound Location Service' click on '+' to bind a Location Service to the created SIP Gateway GW_OBS:

Actions	Screenshot
	 Select the correct Location Service Choose the previously created Location Service LS_OBS (described in previous chapter Configure Location Service)  Leave the option „Disable registration outbound“ unselected (default). Despite this option, no outgoing registration registration will take place, because it is not activated in the selected Location Service.
Enable the SIP Gateway GW_OBS	Finally enable the SIP Gateway: under GW_OBS / Gateway State, check the box 'Enable'. Note that straight after enabling, the Smartnode's SIP-Trunk starts listening to incoming SIP messages. But the SIP signaling will work as planned only after the whole configuration is finished. You may disable the SIP Gateway during the whole configuration process and enable it again at the very end.

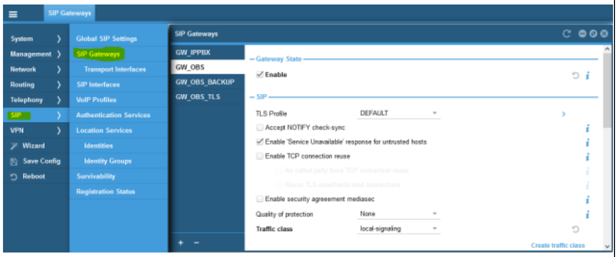
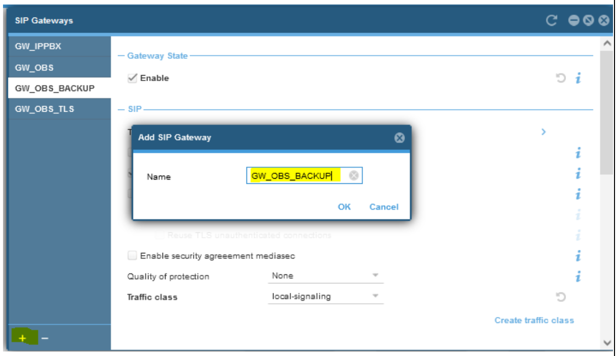
<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>
</td></tr>
<tr><td></td><td>Via CLI:
<pre>
context sip-gateway GW_OBS
  bind location-service LS_OBS
  traffic-class local-signaling

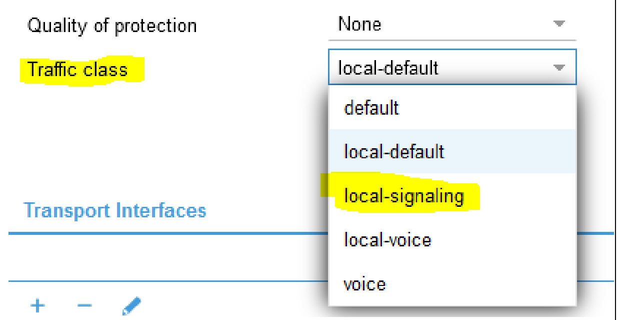
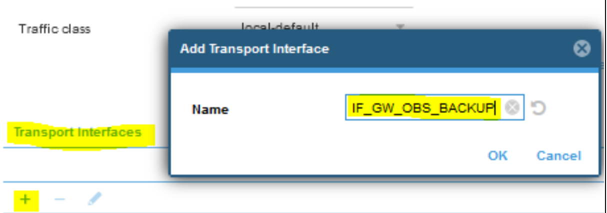
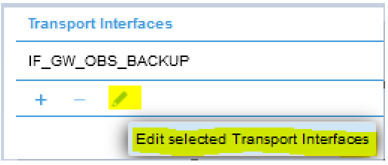
  interface IF_GW_OBS
    transport-protocol udp+tcp 5060
    no transport-protocol tls
    bind ipaddress ROUTER WAN WAN_IP1

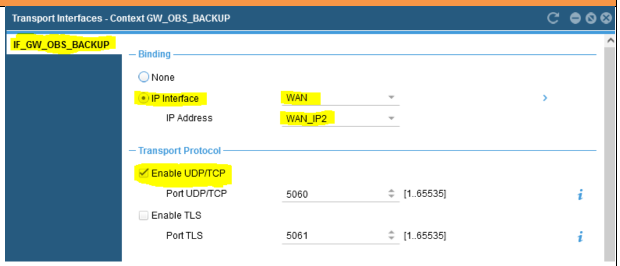
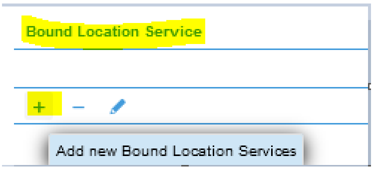
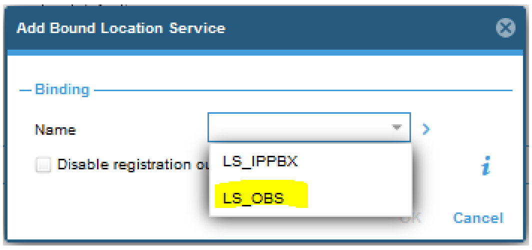
context sip-gateway GW_OBS
  no shutdown
</pre></td></tr>
</table>

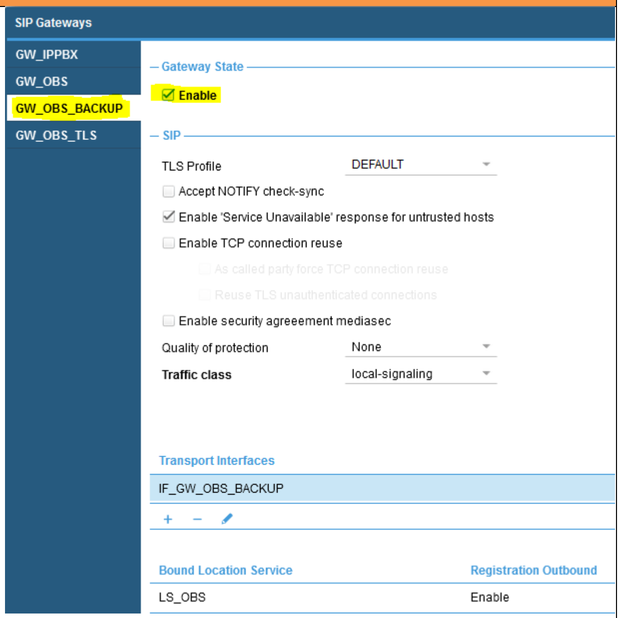
SIP-Gateway / backup local IP-address

This is exactly the same proceeding as for the nominal SBC and it is optional, to be used only for local IP resilience of the eSBC.
only with corresponding naming change including '…_BACKUP' in the gateway name.

Actions	Screenshot
Access the SIP Gateway menu	Via Web UI: Open the menu SIP > SIP Gateways 
Create the SIP Gateway GW_OBS_BACKUP	Click on '+' at the bottom left to create a new SIP Gateway, enter the name 'GW_OBS_BACKUP' and confirm with OK. 
Select the correct traffic class for DSCP tagging for SIP signaling	Select the traffic class 'local-signaling'. Important: this traffic class has been configured in the DSCP profile (profile service-policy SP_WAN_OUT), as explained under Global configuration in the chapter DSCP profile, must be selected here in order to ensure the corresponding packets tagging of the outgoing SIP messages towards the trunk to BTIP/BTalk. Leaving the default setting here would mean no DSCP tagging, so don't miss this part.

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>
</td></tr>
<tr><td>Create transport interface</td><td>As any other name of variable (by convention in capital letters), you are also free to define a name for the transport interface inside the SIP Gateway. It is through this interface that the binding with an IP address from the context IP / Interface is done. By convention we set the name IP_GW_OBS_BACKUP:

</td></tr>
<tr><td>Edit the the created transport interface</td><td>Edit the settings of the newly created transport interface:

</td></tr>
<tr><td>Modify the default settings inside the transport interface</td><td>Under binding select 'IP interface' and select the existing IP interface 'WAN' and select the IP address 'WAN_IP2' (backup WAN IP address of the SmartNode eSBC), both created previously in the Context IP.
Under 'Transport Protocol' check the box 'Enable UDP/TCP'. Leave the default port setting to 5060.</td></tr>
</table>

Actions	Screenshot
	
Bind the location service LS_OBS	Under 'Bound Location Service' click on '+' to bind a Location Service to the created SIP Gateway GW_OBS:  Select the correct Location Service Choose the previously created Location Service LS_OBS (described in previous chapter [Configure Location Service](#))  Leave the option „Disable registration outbound“ unselected (default). Despite this option, no outgoing registration registration will take place, because it is not activated in the selected Location Service.
Enable the SIP Gateway GW_OBS_BACKUP	Finally enable the SIP Gateway: under GW_OBS_BACKUP / Gateway State, check the box 'Enable'. Note that straight after enabling, the Smartnode's SIP-Trunk starts listening to incoming SIP messages. But the SIP signaling will work as planned only after the whole configuration is finished. You may disable the SIP Gateway during the whole configuration process and enable it again at the very end.

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td></td></tr>
<tr><td></td><td>Via CLI:

<code>context sip-gateway GW_OBS_BACKUP</code>
<code> bind location-service LS_OBS</code>
<code> traffic-class local-signaling</code>

<code> interface IF_GW_OBS_BACKUP</code>
<code>   transport-protocol udp+tcp 5060</code>
<code>   no transport-protocol tls</code>
<code>   bind ipaddress ROUTER WAN WAN_IP2</code>

<code>context sip-gateway GW_OBS</code>
<code> no shutdown</code></td></tr>
</table>

5.2.3 Configure VoIP Profiles

The VoIP Profile defines media codecs that will be used by the SIP Interfaces.

The VoIP profile is used to define the codec order in the SDP offer and/or to modify the order or preference in the codec list.

The SIP > VoIP Profiles menu path allows you to specify the individual voice and fax compression codecs and their associated settings for inclusion in a codec list. Different codecs provide varying levels of compression, allowing one to reduce bandwidth requirements at the expense of voice quality.

We will configure two VoIP profiles, one for BTIP and another for BTalk, so that they can be adapted for both SIP-Trunks independently.

The VoIP Profile must be compliant with OBS requirement bellow

- ✓ DTMF via RFC 2833/4733

Note:
For DTMF, the Patton SBC (SN model containing DSP) will be able to convert SIP INFO message to RFC2833/4733.

The SBC supports the RFC 6086 ‘Session Initiation Protocol (SIP) INFO Method and Package Framework’ so it can handle SIP INFO messages carrying DTMF on private IPBX side.

VoIP Profile for BTIP

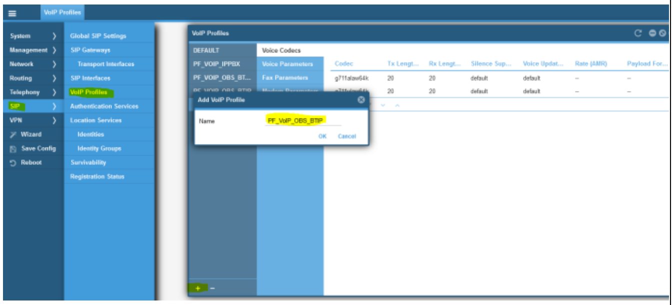
- ***G.722 (If only used)***
- ***G.711 A-law 20 ms***
- ***G.729 20 ms (annexb = no).***

VoIP Codec Profile specific to Orange BTIP:

Description	Codec	Payload Size	Comments
G.722	G.722	20 ms	
Default G711A	G.711 A-Law	20 ms	
G.729	G.729	20 ms	Annex b=No not supported by default

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td>Create VoIP profile PF_VOIP_OBS_BTIP</td><td>Via Web UI:

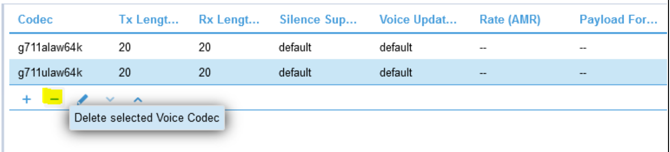
Open the menu SIP > VoIP Profile , then click on ‘+’ button to create a new VoIP profile and enter the name PF_VOIP_OBS_BTIP.


</td></tr>
<tr><td>Proceed to audio codec modifications inside the profile</td><td>By default, a newly created VoIP profile has following two codecs defined in this order:

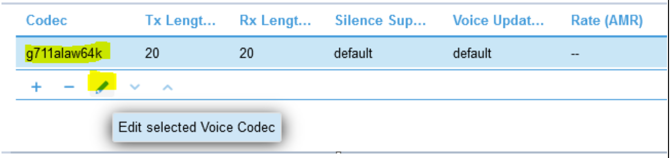
1. G.711 A-law 20 ms

2. G.711 µ-law 20 ms</td></tr>
</table>

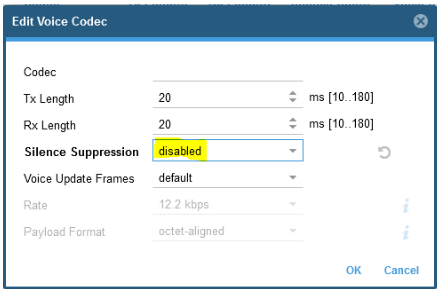
<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>
To remove G.711 µ-law from the codec list just select it and click on '-' button. Additionally, the arrow buttons under the list are used to modify the codec order after you have added all required codecs.



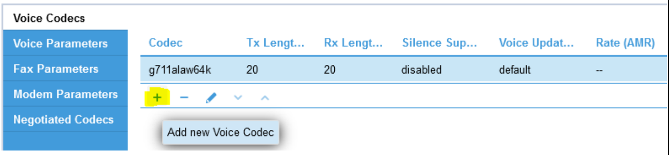
Click on the codec G.711alaw64k, then on the pencil (edit) button to edit parameters of the codec:



In the 'Edit Voice Codec' window disable Silence Suppression in the dropdown list (not supported by BTIP/BTalk SIP-Trunks), which is enabled by default on each created codec. Leave the other parameters to the default values, i.e. Tx Length and Rx Length to 20ms and Voice Update Frames to default, which is disabled by default if Silence Suppression is disabled (Voice Update Frames can be effectively enabled only if Silence Suppression is enabled). Confirm the changes with OK.

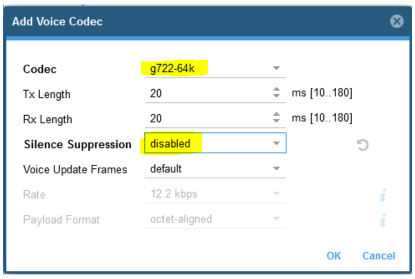


Click on '+' to add the audio codec G.722:

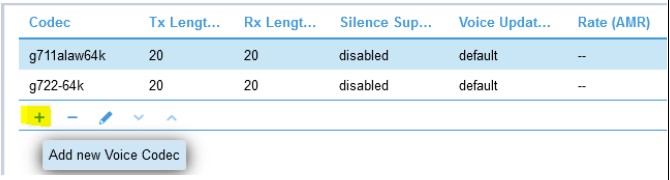


</td></tr>
</table>

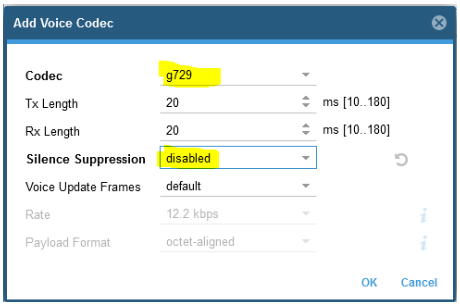
<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>
In the ‘Add Voice Codec’ window, select ‘g722-64k’ in the dropdown codec list, set Silence Suppression to disabled and confirm the changes with OK.



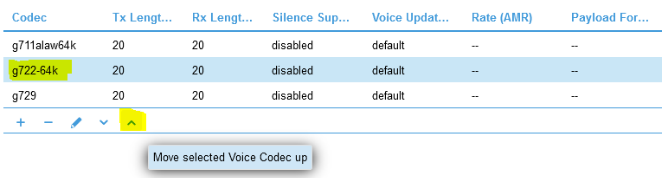
In the modified codec list, click on ‘+’ to also add the audio codec G.729:



In the ‘Add Voice Codec’ window, select ‘g729’ in the dropdown codec list, set Silence Suppression to disabled and confirm the changes with OK.



If you want to set G.722 as first SDP offer in the codec list of the eSBC, then select it and use the arrow Up button:



</td></tr>
</table>

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>Result:
<table>
<tr><th>Codec</th><th>Tx Lengt...</th><th>Rx Lengt...</th><th>Silence Sup...</th><th>Voice Updat...</th><th>Rate (AMR)</th><th>Payload For...</th></tr>
<tr><td>g722-64k</td><td>20</td><td>20</td><td>disabled</td><td>default</td><td>--</td><td>--</td></tr>
<tr><td>g711alaw64k</td><td>20</td><td>20</td><td>disabled</td><td>default</td><td>--</td><td>--</td></tr>
<tr><td>g729</td><td>20</td><td>20</td><td>disabled</td><td>default</td><td>--</td><td>--</td></tr>
</table>
+ −
Note : The default setting for G.729 codec is without Annex B, so an attribute line “annexb=no” will be always be present for G729 in SDP.

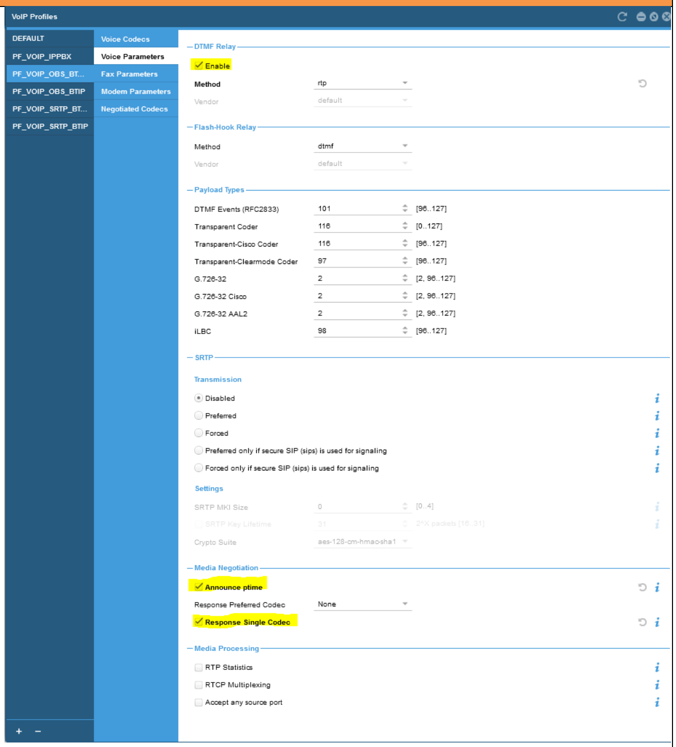
If another order is preferred, just use the same logic as above to modify the list.</td></tr>
<tr><td>Modify Voice Parameters</td><td>Under the same VoIP profile, click on the next submenu ‘Voice Parameters’ and modify only these three parameters from their default values (and leave all the other parameters unchanged):

• DTMF Relay: check the box ‘enable’ and set the method to RTP, in order to use RTP payload for DTMF digits (RFC 2833/4733)

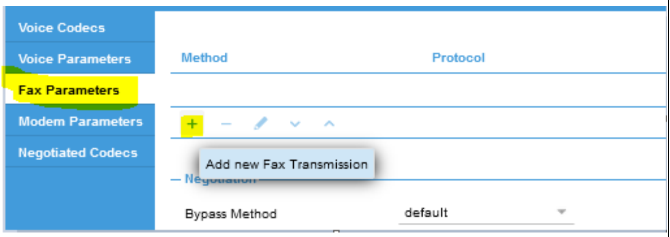
• Media Negotiation:

o Check the box ‘Announce ptime’ to add the attribute a=ptime:20 in the SDP of SIP messages sent by the eSBC.

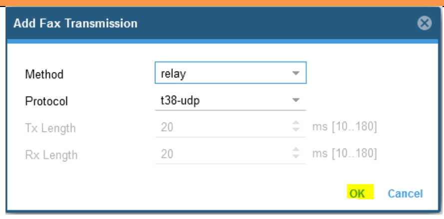
o Check the box ‘Response Single Codec’ to transmit only the negotiated codec in SDP of 200 OK responses instead of the codec list.</td></tr>
</table>

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>
</td></tr>
<tr><td>Add T38 Fax relay</td><td>Under the same VoIP profile, click on the next submenu ‘Fax Parameters’ in order to add the T38 Fax relay capability.

Click on ‘+’ to add new fax transmission type:



In the ‘Add Fax Transmission’ window just leave the default selections ‘relay’ and ‘t38-udp’ and confirm:</td></tr>
</table>

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>

Remark: optionally an additional bypass method with G.711 may be added, if required, but this is not supported by OBS BTIP/BTalk.

Patton eSBCs transparently transmit the T38 fax relay in pass-through mode between the ATA device (with fax machine) and the SIP-Trunk, from one leg to the other, meaning that they are not able to transcode, for example between G711 and T38. Patton analog Gateways / ATAs in contrary are able to terminate T38.

Both Patton eSBCs and Gateways support Fax G3 standard over T38, with speeds of up to 14400 kbits/s and typically operate at 9600 bits/s. Super G3 can only be supported in conjunction with the bypass method with G.711 (see above). G.711 bypass for T38 should be disabled for OBS BTIP/BTalk. In this case only G3 with speeds up to 14400 kbits are supported, without Fallback capability.</td></tr>
<tr><td>Enable Codec Negotiation</td><td>Important: SIP protocol offers a codec negotiation mechanism. It is not guaranteed that the first codec in the SDP list will be used to set up the connection. Each codec in the list may be used.

On Patton eSBCs the codec negotiation is disabled by default in the VoIP profile, which honors the codec lists from each call leg independently, formed out of the remote and local capabilities. On HW DSP-based eSBC models, the DSP is inserted into the RTP path to make sure each side can use its codec. If necessary, the DSP will transcode between the codecs of the two RTP streams. Enabled “codec negotiation” will keep the DSP out of the picture in established calls and tries to negotiate a common codec for both call legs. We recommend to enable “codec negotiation” only on SN-models without HW DSP processors (SN500, vSN, SN5301 … see details in the list of the certified product versions)

Configurable only via CLI:

`profile voip PF_VOIP_OBS_BTIP`
`  codec negotiation`</td></tr>
</table>

Configuration method of the same complete VoIP Profile via CLI:

```
profile voip PF_VOIP_OBS_BTIP
  codec 1 g722-64k rx-length 20 tx-length 20
  codec 2 g711alaw64k rx-length 20 tx-length 20
  codec 3 g729 rx-length 20 tx-length 20
  codec negotiation                  <-------- only on models without HW
DSP
  dtmf-relay rtp
  sdp-ptime-announcement
  codec response single
  fax transmission 1 relay t38-udp
```

VoIP Profile for BTalk

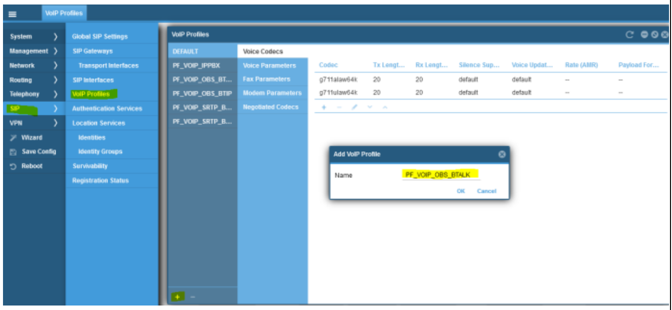
- ***G.711 A-law 20 ms or G.711 µ-law 20 ms***
- ***G.729 20 ms (annexb = no).***

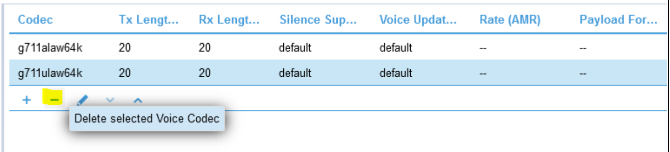
Note:

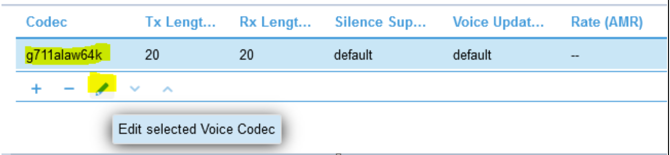
G.711 µ-law 20 ms can be requested by OBS, specifically on demand. If this is the case, it should just be added to the codec list in this VoIP profile.

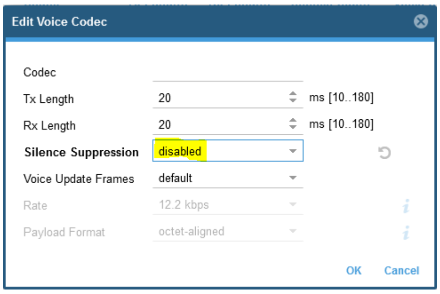
VoIP Codec Profile specific to Orange BTIP:

Description	Codec	Payload Size	Comments
Default G711A	G.711 A-Law	20 ms	
Default G711µ	G711 U-Law	20 ms	Optional on request
G.729	G.729	20 ms	

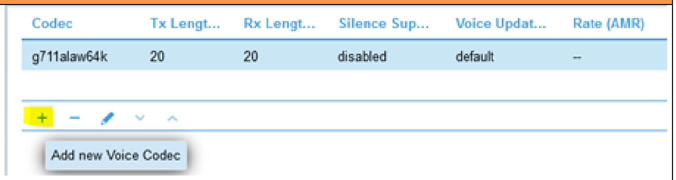
Actions	Screenshot
Create VoIP profile PF_VOIP_OBS_BTALK	Via Web UI: Open the menu SIP > VoIP Profile , then click on '+' button to create a new VoIP profile and enter the name PF_VOIP_OBS_BTALK. 

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td>Proceed to audio codec modifications inside the profile</td><td>
By default, a newly created VoIP profile has following two codecs defined in this order:
3. G.711 A-law 20 ms
4. G.711 µ-law 20 ms
To remove G.711 µ-law (if not required) from the codec list just select it and click on ‘-‘ button. Additionally, the arrow buttons under the list are used to modify the codec order after you have added all required codecs.


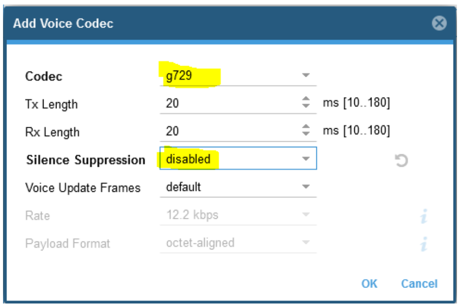
Click on the codec G.711alaw64k, then on the pencil (edit) button to edit parameters of the codec:


In the ‘Edit Voice Codec’ window disable Silence Suppression in the dropdown list (not supported by BTIP/BTalk SIP-Trunks), which is enabled by default on each created codec. Leave the other parameters to the default values, i.e. Tx Length and Rx Length to 20ms and Voice Update Frames to default, which is disabled by default if Silence Suppression is disabled (Voice Update Frames can be effectively enabled only if Silence Suppression is enabled). Confirm the changes with OK.


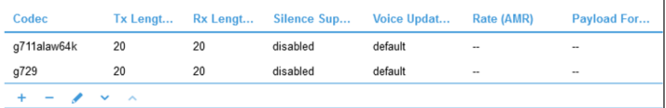
In the modified codec list, click on ‘+’ to add the audio codec G.729:
</td></tr>
</table>

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>


In the ‘Add Voice Codec’ window, select ‘g729’ in the dropdown codec list, set Silence Suppression to disabled and confirm the changes with OK.



Result:

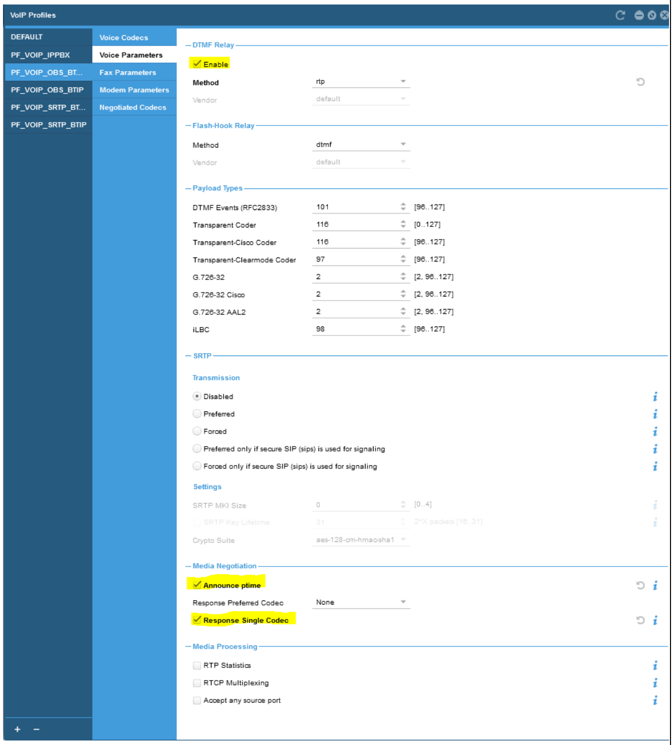


If another order is preferred, just select one codec and use the arrow buttons to modify the order. Also if the optional codec G.711 µ-law is required, add it (and adapt the order) using the same logic.
</td></tr>
<tr><td>Modify Voice Parameters</td><td>
Under the same VoIP profile, click on the next submenu ‘Voice Parameters’ and modify only these two parameters from their default values (and leave all the other parameters unchanged):

- DTMF Relay: check the box ‘enable’ and set the method to RTP, in order to use RTP payload for DTMF digits (RFC 2833/4733)
- Media Negotiation:
  - Check the box ‘Announce ptime’ to add the attribute a=ptime:20 in the SDP of SIP messages sent by the eSBC.
</td></tr>
</table>

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>

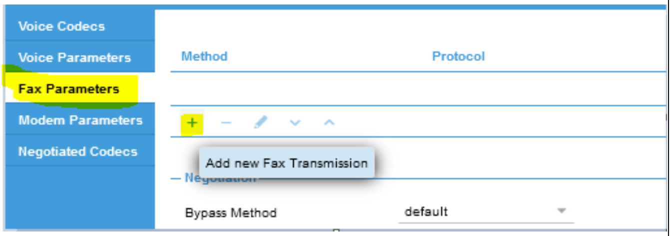
- Check the box 'Response Single Codec' to transmit only the negotiated codec in SDP of 200 OK responses instead of the codec list.



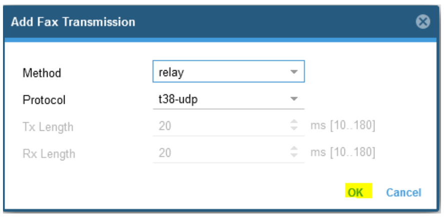
</td></tr>
</table>

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td>Add T38 Fax relay</td><td>
Under the same VoIP profile, click on the next submenu ‘Fax Parameters’ in order to add the T38 Fax relay capability.

Click on ‘+’ to add new fax transmission type:



In the ‘Add Fax Transmission’ window just leave the default selections ‘relay’ and ‘t38-udp’ and confirm:



Remark: optionally an additional bypass method with G.711 may be added, if required, but this is not preferred for OBS BTIP/BTalk.

Patton eSBCs transparently transmit the T38 fax relay in pass-through mode between the ATA device (with fax machine) and the SIP-Trunk, from one leg to the other, meaning that they are not able to transcode, for example between G711 and T38. Patton analog Gateways / ATAs in contrary are able to terminate T38.
Both Patton eSBCs and Gateways support Fax G3 standard over T38, with speeds of up to 14400 kbits/s and typically operate at 9600 bits/s.
Super G3 can only be supported in conjunction with the bypass method with G.711 (see above). G.711 bypass for T38 should be disabled for OBS BTIP/BTalk. In this case only G3 with speeds up to 14400 kbits are supported, without Fallback capability.
</td></tr>
<tr><td>Enable Codec Negotiation</td><td>
Important: SIP protocol offers a codec negotiation mechanism. It is not guaranteed that the first codec in the SDP list will be used to set up the connection. Each codec in the list may be used.

On Patton eSBCs the codec negotiation is disabled by default in the VoIP profile, which honors the codec lists from each call leg independently, formed out of the remote and local capabilities. On HW DSP-based eSBC
</td></tr>
</table>

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>models, the DSP is inserted into the RTP path to make sure each side can use its codec. If necessary, the DSP will transcode between the codecs of the two RTP streams. Enabled “codec negotiation” will keep the DSP out of the picture in established calls and tries to negotiate a common codec for both call legs. We recommend to enable “codec negotiation” only on SN-models without HW DSP processors (SN500, vSN, SN5301 … see details in the list of the certified product versions)

Configurable only via CLI:

<pre>profile voip PF_VOIP_OBS_BTIP
  codec negotiation</pre></td></tr>
<tr><td></td><td>Configuration method of the same complete VoIP Profile via CLI:

<pre>profile voip PF_VOIP_OBS_BTALK
  codec 1 g722-64k rx-length 20 tx-length 20
  codec 2 g711alaw64k rx-length 20 tx-length 20
  codec 3 g729 rx-length 20 tx-length 20
  codec negotiation   <--------- only models without HW DSP
  dtmf-relay rtp
  sdp-ptime-announcement
  codec response single
  fax transmission 1 relay t38-udp</pre></td></tr>
</table>

5.2.4 Configure SIP Interfaces

Orange BTalk / BTIP UDP

Patton eBSC will be configured to be compliant with Orange BTalk/BTIP specification:

- ✓ For unencrypted BT SIP Trunk architecture, we need to configure UDP port 5060
- ✓ For SIP-Trunk keep alive done with “Options” message (every 300 seconds)
- ✓ 2 SIP Gateways will be configured for local redundancy purpose.

Commenté [CSO4]: Please adapt those sentences to Patton eSBC Sip Trunk resiliency configuration

SIP Profile must be configured to be compliant with Orange BTalk/BTIP specifications:

- ✓ Session Timer is not supported

The mentioned parameters in the tables below are the one specific to Orange Profile. All the other parameters must be left as «default value».

SIP Interface	Host FQDN/IP	Port	Protocol	Transport
1	<BT_Nominal_IP>	5060	UDP	Monitor: Sip Options Keep Alive Frequency: 300
2	<BT_Backup_IP>	5060	UDP	Monitor: Sip Options Keep Alive Frequency: 300

Note:

IP's set in the "Host IP" are the one's provided by Orange for the BTalk/BTIP SIP trunk. "Options" message will be sent by the Patton SBC to verify if the Orange BTalk/BTIP network is reachable.
All the screenshots below showing some IP address are given as example. You should replace them by the correct IP or FQDN in your context.

Note2: *In case of SIP Provisional Response ACKnowledgement (PRACK RFC 3262) could be required (such as for Cisco CUCM) to be interworked with Orange which does not support PRACK.* PRACK to Early Offer conversion: Patton eSBC is compliant with the required behavior of OBS as long as its default configuration is used, i.e. as long as PRACK is not enabled on the SIP interface towards OBS. No specific conversion or specific command on the OBS SIP Interface is needed. By default SN eSBC never sends delayed offers (INVITE without SDP), unless explicitly configured. eSBC is configured by default to send INVITE without 100rel tag.

Note3: As shown in the chapter Objects in the chapter Patton Global Configuration, the configuration element SIP Gateway is the main internal interface between the Call Router and the corresponding IP address through which the Smartnode is communicating with the remote side.

It is in the SIP Interface configuration part of the Context CS that the main SIP signaling parameters towards remote peer are defined. By design one SIP Interface per remote peer should be created.

We assign it an explicit name depending on the remote side that Patton eSBC will communicate with through that SIP Interface.

Example: IF_SIP_OBS_BTIP_MAIN for the interface communicating with BTIP nominal (main) SBC. SIP Interface configuration contains specific parts related either to BTIP/BTalk (SIP/UDP) or to BTIPoI/BToI (SIP/TLS) because the complete configuration is specific to the SIP-Trunk requirements in terms of transport protocol used (UDP or TLS) and media/VoIP profile used.

Main parameters defined in a SIP Interface:

- Context SIP Gateway binding (link to the corresponding IP interface / IP address)
- Internal Call Routing for calls received via this SIP Interface
- Remote peer's IP/FQDN and port number
- Local domain name or IP address to be used in the host part of the From header in outgoing SIP requests
- Applied VoIP profiles (i.e. media codec profiles)
- Applied SIP tunneling profiles for specific manipulations
- Various header customization parameters (see corresponding chapter)
- Call handling methods: forwarded calls (use REFER or re-INVITE), transferred calls, Early-Media handling
- Uri-scheme to be used (sip or sips)
- Enabling / disabling handling of call privacy (Privacy and PAI/PPI headers)
- Session refresh method and periodicity

- SIP Options ping periodicity
- Transport protocol

The corresponding configuration menu is accessible under *SIP > SIP Interfaces* menu path in the Web GUI, or under “context cs > interface sip” CLI configuration element.

The SIP Interface parameters that are listed in the table below are only the non-default necessary parameters and values that shall be configured to respect Orange Certified Border specs.
All the other parameters of the SIP Interface configuration must be left at their respective default values.

Description	Parameter	Value	Comments
SIP call hold method to be used. Default setting: zero-IP, to be configured to the preferred method sendonly.	hold-method	direction-attribute sendonly	
Early Media handling according to RFC5009	early-media	accept authorized	
Do not accept incoming transferred calls from OBS with REFER method	call-transfer accept	no call-transfer accept	
Support REFER to re-INVITE conversion towards OBS	call-transfer emit	no call-transfer emit	REFER to Re-INVITE :When Blind and Consultative transfer are handled by the SIP REFER method, the SBC will generate a Re-INVITE towards the transfer target
Enable support of privacy and PAI/PPI headers	privacy	(just enable privacy)	
Apply the correct VoIP (media) profile	use profile voip	PF_VOIP_OBS_BTIP (*) PF_VOIP_OBS_BTALK (*) (*): VoIP profile definition – see in corresponding chapter	
OBS-specific header manipulation required in order to achieve the concatenated header content for User-Agent or Server header sent from eSBC (<IPBX Vendor v.X.X + SBC vendorV.X.X>)	use profile sip-tunneling out	OBS_USER_AGENT_CONCAT (*) (*): sip-tunneling profile definition - see in corresponding chapter	
Keep-alive OPTIONS	penalty-box sip-option-trigger	interval 300 timeout 300 force udp	
Session refresh method	session-timer	session-timer 1800 method update	

Design concept used for the resilience:

We consider the following inputs

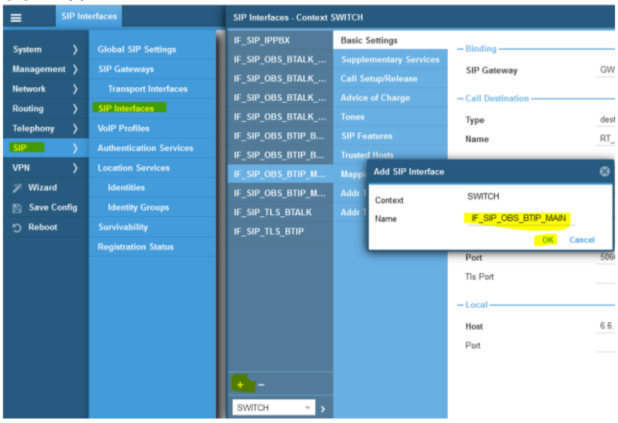
- On OBS infrastructure side for unencrypted SIP / UDP there are totally four SBCs (two pairs):
  - A pair of SBCs (Nominal & Backup) for BTIP: in our example with IP addresses <BTIP_Nominal_IP> & <BTIP_Backup_IP> respectively
  - A pair of SBCs (Nominal & Backup) for BTalk: in our example with IP addresses <BTalk_Nominal_IP> & <BTalk_Backup_IP> respectively.
- Optional :On Patton sSBC side two IP addresses can be configured for IP address resilience purposes. In our example these are the IP addresses 6.6.77.10 and 6.6.77.11
- On Patton eSBC each SIP Interface configuration contains by design a local and a remote host for proper signaling, that it will set into the host part of From header (local) and the host part of the To header (remote).

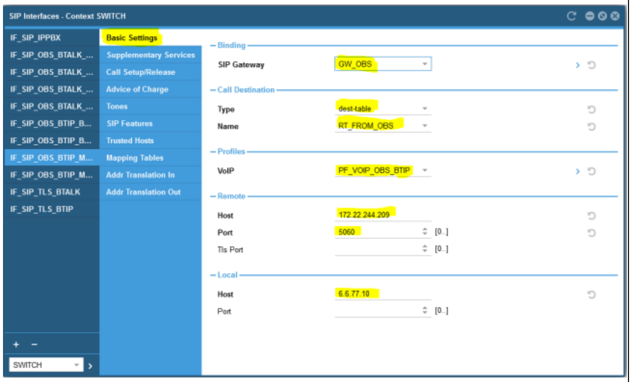
This generates the following 8 combinations below with respective SIP logical Interface names used for resiliency purpose (Hunt group §) :

	BTIP		BTalk	
	Nominal SBC	Backup SBC	Nominal SBC	Backup SBC
Patton eSBC Nominal IP	IF_SIP_OBS_BTIP_MAIN	IF_SIP_OBS_BTIP_BACKUP	IF_SIP_OBS_BTALK_MAIN	IF_SIP_OBS_BTALK_BACKUP
Patton eSBC Backup IP	IF_SIP_OBS_BTIP_MAIN_11	IF_SIP_OBS_BTIP_BACKUP_11	IF_SIP_OBS_BTALK_MAIN_11	IF_SIP_OBS_BTALK_BACKUP_11

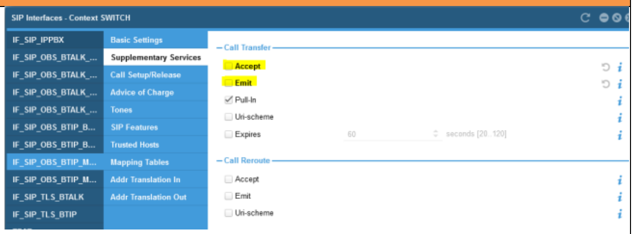
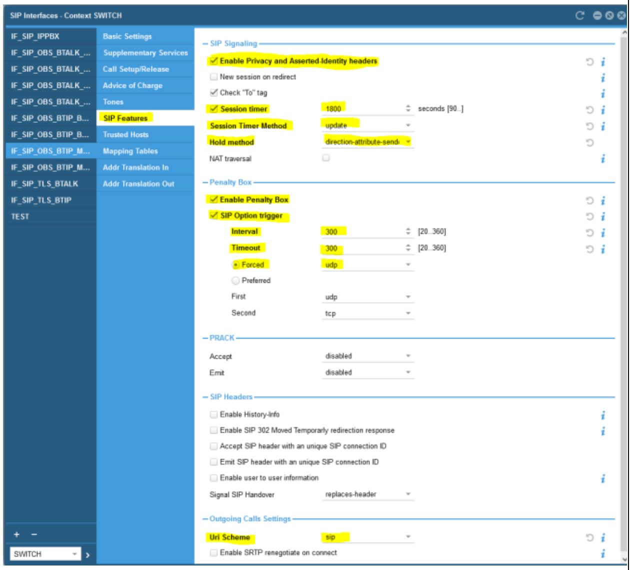
The configuration of the 8 SIP Interfaces listed in this table are very similar. Only a few parameters differ: local / remote hosts, SIP Gateway (main/backup) and VoIP media profile.

In order to simplify the guidelines and not repeat the same description for all the SIP Interfaces, we will describe the configuration of all the 8 SIP Interfaces shown above only once by precising the different specific values that must be entered for each of them.

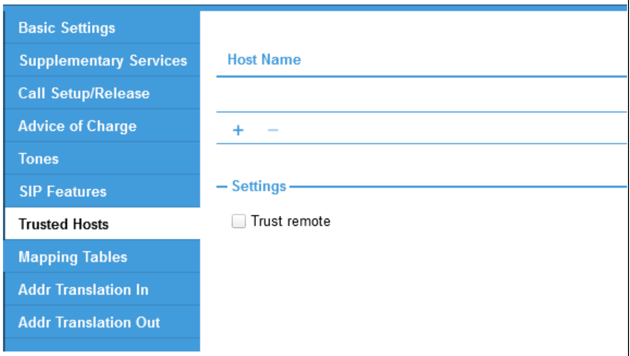
Actions	Screenshot
Create SIP Interface	Via Web UI: Open the menu SIP > SIP Interfaces, then click on '+' at the bottom left to create a new SIP Interface. Insert the SIP interface name according to the local and remote peer: IF_SIP_OBS_BTIP_MAIN IF_SIP_OBS_BTIP_BACKUP IF_SIP_OBS_BTALK_MAIN IF_SIP_OBS_BTALK_BACKUP And optionally (if IP local resilience is used): IF_SIP_OBS_BTIP_MAIN_11 IF_SIP_OBS_BTIP_BACKUP_11 IF_SIP_OBS_BTALK_MAIN_11 IF_SIP_OBS_BTALK_BACKUP_11 Note: the ending extension _11 is just a conventional marker in the name, because in the test lab the local backup IP address was 6.6.77.11. 
Configure basic settings of the SIP Interface	Select the submenu 'Basic Settings'. Select the following created variables and insert the following values: SIP Interface IF_SIP_OBS_BTIP_MAIN: • Binding / SIP Gateway: choose the previously created SIP Gateway GW_OBS (see chapter SIP-Gateway towards nominal BTIP/BTalk SBC) • Call Destination: Type -> select 'dest-table' ; Name -> select RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX)

Actions	Screenshot
	• Profiles / VoIP: select the previously created VoIP profile PF_VOIP_OBS_BTIP (see VoIP Profile for BTIP) • Remote: Host -> enter the IP address of the main BTIP SBC / <BTIP_Nominal_IP>; Port -> remote UDP listen port (5060) • Local: Host -> enter the main IP address configured on the eSBC WAN interface towards OBS.  By proceeding the same way as for the SIP Interface IF_SIP_OBS_BTIP_MAIN, select the other seven SIP Interfaces listed in the previous table above and go through the same submenu 'Basic settings' by setting / entering the following values: • Interface IF_SIP_OBS_BTIP_BACKUP Binding / SIP Gateway: GW_OBS Call Destination: Type -> 'dest-table' ; Name -> RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX) Profiles: VoIP -> PF_VOIP_OBS_BTIP Remote: Host -> BTIP Backup <BTIP_Backup_IP>; Port -> 5060 Local: Host -> eSBC Main 6.6.77.10 • Interface IF_SIP_OBS_BTALK_MAIN Binding / SIP Gateway: GW_OBS Call Destination: Type -> 'dest-table' ; Name -> RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX) Profiles: VoIP -> PF_VOIP_OBS_BTALK Remote: Host -> BTalk Main <BTalk_Nominal_IP>; Port -> 5060 Local: Host -> eSBC Main 6.6.77.10 • Interface IF_SIP_OBS_BTALK_BACKUP Binding / SIP Gateway: GW_OBS Call Destination: Type -> 'dest-table' ; Name -> RT_FROM_OBS

Actions	Screenshot
	(see chapter Routing Table from OBS to IPPBX) Profiles: VoIP -> PF_VOIP_OBS_BTALK Remote: Host -> BTalk Backup <BTalk_Backup_IP>; Port -> 5060 Local: Host -> eSBC Main 6.6.77.10
Optional : In case of resiliency, you will have to configure additional SIP interfaces (flagged with _11 in this example) targeting OBS SIP terminations within a different local eSBC IP address	• Interface IF_SIP_OBS_BTIP_MAIN_11 Binding / SIP Gateway: GW_OBS_BACKUP Call Destination: Type -> 'dest-table' ; Name -> RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX) Profiles: VoIP -> PF_VOIP_OBS_BTIP Remote: Host -> BTIP Main <BTIP_Nominal_IP>; Port -> 5060 Local: Host -> eSBC Backup 6.6.77.11 • Interface IF_SIP_OBS_BTIP_BACKUP_11 Binding / SIP Gateway: GW_OBS_BACKUP Call Destination: Type -> 'dest-table' ; Name -> RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX) Profiles: VoIP -> PF_VOIP_OBS_BTIP Remote: Host -> BTIP Backup <BTIP_Backup_IP>; Port -> 5060 Local: Host -> eSBC Backup 6.6.77.11 • Interface IF_SIP_OBS_BTALK_MAIN_11 Binding / SIP Gateway: GW_OBS_BACKUP Call Destination: Type -> 'dest-table' ; Name -> RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX) Profiles: VoIP -> PF_VOIP_OBS_BTALK Remote: Host -> BTALK Main <BTalk_Nominal_IP>; Port -> 5060 Local: Host -> eSBC Backup 6.6.77.11 • Interface IF_SIP_OBS_BTALK_BACKUP_11 Binding / SIP Gateway: GW_OBS_BACKUP Call Destination: Type -> 'dest-table' ; Name -> RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX) Profiles: VoIP -> PF_VOIP_OBS_BTALK Remote: Host -> BTALK Backup <BTalk_Backup_IP>; Port -> 5060 Local: Host -> eSBC Backup 6.6.77.11
Configure supplementary services of each SIP Interface	Select the submenu Supplementary Services in each SIP Interface. Uncheck the boxes Call Transfer Accept and Call Transfer Emit (which are enabled by default) in order to disable these methods:

Actions	Screenshot
	 Proceed to the same modification on all SIP Interfaces towards OBS (see previous list). Meaning of these two parameters: Call Transfer Accept: if enabled, incoming call transfers with REFER method will be accepted; if disabled, incoming call transfers with REFER method will be rejected and incoming call transfers with re-INVITE method will be accepted. Call Transfer Emit: if enabled, outgoing call transfers with REFER method will be sent; if disabled, no outgoing call transfers with REFER method but re-INVITE method will be used towards the SIP-Trunk.
Configure the SIP Features of each SIP Interface	Select the submenu SIP Features and modify the following parameters as described below on each SIP interface. 

Actions	Screenshot
	• Enable Privacy and Asserted-Identity headers: (disabled by default) enable it in order to support sending Privacy and PAI/PPI headers towards the SIP-Trunk in appropriate call scenarios (typically for outgoing anonymous calls) according to RFC3323 and RFC3325. Note that some additional header manipulation is required in order for anonymous calls to work as specified for BTIP and BTalk -> see From, PAI/PPI headers for anonymous calls in the chapter SIP rules & manipulations (SBC Application). • Enable the session timer and configure it to 1800 seconds: the session refresh will be done each 1800 / 2 = 900 seconds (15 minutes). • As session timer method select 'update' in order to use the SIP method Update to refresh long duration calls. • Change the hold method from zero-ip (default) to direction-attribute-sendonly in order set the SDP attribute sendonly on Call Hold. • Enable the Penalty-Box feature: this feature checks the availability of the remote peer. • Enable the SIP Option trigger in order to activate the use of SIP Options Pings in correlation with the enabled Penalty-Box feature. • Set the Interval and Timeout timers to 300 seconds. This is the time interval between two subsequent SIP Options messages sent by the eSBC through this SIP Interface. • Force the use of UDP transport protocol. We use this fix setting instead of the 'preferred' setting which combines UDP and TCP with a preference order, which is not necessary here because of the other interfaces dedicated to SIP/TLS/TCP. • Under Outgoing Calls Settings / URI-scheme, select SIP Change all those parameters the same way on the other seven SIP Interfaces towards OBS (see previous list).
Trusted hosts	Optional, useful for increased level of security, additionally to the ACL lists already used on IP level. A list of trusted remote peers can be configured on SIP interfaces. If configured, only connections with peers in that list will be accepted. The list may contain IP-addresses or FQDNs.

Actions	Screenshot
	In case you would like to use this feature, select the check box ‘Trust remote’ and add the corresponding FQDN / IP-address of the remote peer. 
Address Translation In	See chapter [Diversion header – incoming calls]
Address Translation Out	See chapters [From, PAI/PPI headers for anonymous calls] [Diversion header – outgoing calls]
Enable Early Media support according to RFC5009	Only via CLI: While the SIP dialog is in a provisional state (i.e., when the call is not connected yet), the P-Early-Media header defines with a direction attribute ("sendrecv", "sendonly", "recvonly", or "inactive") if early-media is allowed to be passed-through or if it has to be blocked by the SmartNode. With the new CLI command (SW version 3.20.1 or higher) “early-media accept” the user can specify the early-media processing mode. The behavior of previous SW releases (prior to 3.20.1) is reflected by the option ‘auto’. auto: No P-Early-Media header processing. Early media is accepted as soon as the device receives a provisional SIP response with SDP whose direction attribute allows the transmission. Further provisional SIP responses with SDP may change the current media direction whereas SIP responses without SDP have no effect on the current media direction. authorized: Early media is only accepted if explicitly authorized by the P-Early-Media header. Authorization happens with the P-Early-Media direction attribute ("sendrecv", "sendonly", "recvonly", or "inactive"), which can suppress a media direction that is enabled by SDP at the same time. Once a SIP response with SDP and with a P-Early-Media

Actions	Screenshot
	header has been received, further provisional responses with SDP may change the current media direction as long as they carry a P-Early-Media header as well, whereas SIP responses without SDP have no effect. OBS specification for BTIP / BTalk corresponds to the second option 'authorized', so following CLI is required: **`early-media accept authorized`** Apply this CLI configuration to all SIP interfaces towards BTIP / BTalk.
Whole SIP Interface configuration via CLI	`context cs SWITCH` `interface sip <if_sip_name>` `bind context sip-gateway <sip_gw_name>` `route call dest-table RT_FROM_OBS` `remote <remote_ip> 5060` `local <local_ip>` `hold-method direction-attribute sendonly` `early-media accept authorized` `no call-transfer accept` `no call-transfer emit` `privacy` `uri-scheme sip` `use profile voip <voip_profile>` `penalty-box sip-option-trigger interval 300 timeout 300 force udp` `session-timer 1800 method update`

Duplicate and replace values in brackets with following values depending on the SIP Interface to be configured:

<if_sip_name>	IF_SIP_OBS_BTIP_MAIN	IF_SIP_OBS_BTIP_BACKUP	IF_SIP_OBS_BTALK_MAIN	IF_SIP_OBS_BTALK_BACKUP
<sip_gw_name>	GW_OBS	GW_OBS	GW_OBS	GW_OBS
<remote_ip>	<BTIP_Nominal_IP>	<BTIP_Backup_IP>	<BT_Nominal_IP>	<BT_Backup_IP>
<local_ip>	6.6.77.10	6.6.77.10	6.6.77.10	6.6.77.10
<voip_profile>	PF_VOIP_OBS_BTIP	PF_VOIP_OBS_BTIP	PF_VOIP_OBS_BTALK	PF_VOIP_OBS_BTALK

Optional : In case of resiliency, you will have to configure additional SIP interfaces (flagged with _11 in this example) targeting OBS SIP terminations within a different local eSBC IP address

<if_sip_name>	IF_SIP_OBS_BTIP_MAIN_11	IF_SIP_OBS_BTIP_BACKUP_11	IF_SIP_OBS_BTALK_MAIN_11	IF_SIP_OBS_BTALK_BACKUP_11
<sip_gw_name>	GW_OBS_BACKUP	GW_OBS_BACKUP	GW_OBS_BACKUP	GW_OBS_BACKUP
<remote_ip>	<BTIP_Nominal_IP>	<BTIP_Backup_IP>	<BT_Nominal_IP>	<BT_Backup_IP>
<local_ip>	6.6.77.11	6.6.77.11	6.6.77.11	6.6.77.11
<voip_profile>	PF_VOIP_OBS_BTIP	PF_VOIP_OBS_BTIP	PF_VOIP_OBS_BTALK	PF_VOIP_OBS_BTALK

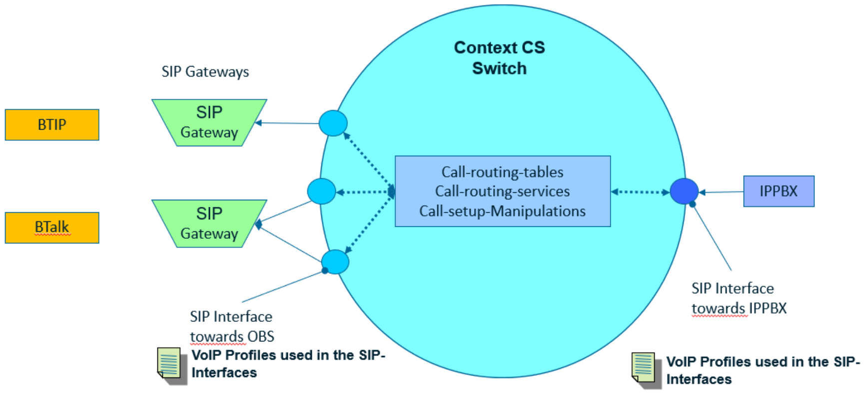
IPPBX

We mention here only the parameter, which is relevant for the local ring-back tone generation towards IPPBX, when the provisional 180 Ringing response from BT/BTIP SIP-Trunk is either without SDP or with SDP and without P-Early-Media header, according to RFC3960, RFC5009 and the technical specifications for OBS BTIP / BTalk.

Actions	Screenshot
Enable local RBT generation towards IPPBX	Only via CLI: **`interface sip IF_SIP_IPPBX`** **`  early-media emit forced`**

5.2.5 Configure Call Routing

The internal Call Routing functionality of the eSBC provides a powerful and flexible routing configuration inside the Context CS Switch. The working principle is displayed in the diagram below.



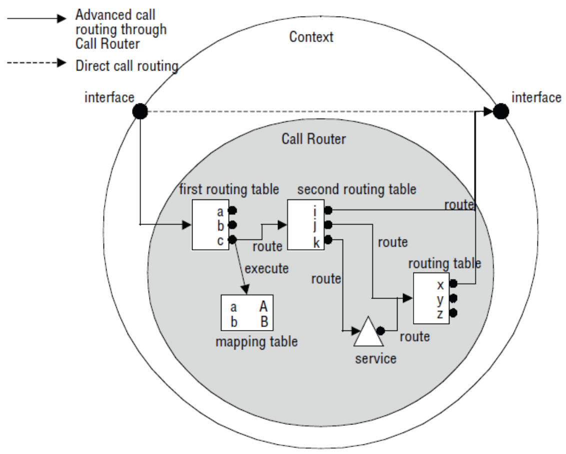
The call routing can be configured to route calls directly from one SIP Interface to another, or through routing tables or through routing services (such as Hunt-Group). Different combinations are possible in order to meet the exact requirements of the customer setup.
Additionally, mapping tables can be called by each routing table to perform different kinds translation rules regarding called/calling number manipulation, called/calling SIP URI, called/calling type of number etc.

This chapter provides the minimum needed configuration to route calls between the SIP Interfaces facing BTIP/BTalk SIP-Trunk and the SIP Interface facing the IPPBX. You could be invited to customize them according to your own requirements.

Example of the naming convention used in the following objects and parameters explanations. Note that it is not mandatory, so you are free to use your own naming convention:

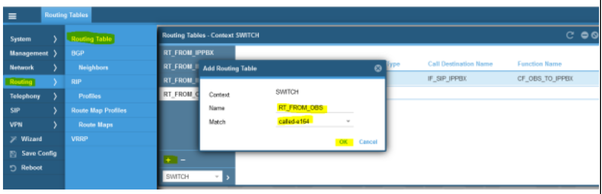
- Routing table names in our examples always start with RT_...
  Example: RT_FROM_OBS is the routing table used to route all incoming calls originating from OBS SIP-Trunk (BTIP or BTalk)
- Mapping table names always start with MT_...
  Example: MT_IPPBX_TO_OBS_CDPN designates a mapping table performing a manipulation of the called party number. CDPN means CalleD Party Number. CNPN would mean CalliNg Party Number, but you can use any other convenient naming convention like A_NUM or B_NUM suffix, or similar, at the end of the mapping table name.
- Some routing tables need to call more than just one mapping table, for example one for calling party number and another for called party number manipulation. In such cases they are grouped in a Complex Function which executes them one by one in a configured order. The suggested naming convention for complex functions is CF_...
  Example: CF_IPPBX_TO_OBS designates the complex function that will be applied to the routing table RT_FROM_IPPBX, i.e. to the call direction from IPPBX to OBS SIP-Trunk.

The following figure represents the main working principle of the direct call routing, from one SIP interface to another, and the one of the advanced call routing described in this guide, which goes through a routing table and optionally calls a mapping table for manipulation purposes.

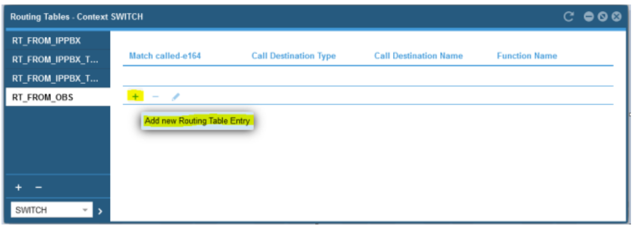


Routing Table from OBS to IPPBX

Incoming calls from OBS are received through one of the SIP Interfaces facing OBS SIP-Trunk (see Configure SIP Interfaces). We will name it RT_FROM_OBS according to the naming convention suggested above.

Actions	Screenshot
Create routing table RT_FROM_OBS	Via Web UI: Open the menu Routing > Routing Table, then click on '+' to create a new call routing table. • Enter the name 'RT_FROM_OBS' and confirm with OK. • Under 'match' enter the matching type of the routing table, in this case 'called-e164'. 

<table>
<tr>
<td>Create a table entry in RT_FROM_OBS</td>
<td>In the newly created routing table, create a new routing table entry by clicking on the '+' button under the table list:



In the next window select the following settings:
Match called-e164: select 'Default'. This means that the route will be chosen for any called e164 number (default call route in this table).
Call Destination: Type -> select 'dest-interface'. This means that the destination of the route will be an interface. In our scenario (pure IP-IP eSBC with SIP and no TDM interfaces) this will be a previously configured SIP Interface. On hybrid eSBCs this could also be an ISDN or analog interface.
Name: select the SIP Interface 'IF_SIP_IPPBX' facing the IPPBX on the LAN side.
Function to apply: leave 'None' if no number manipulation is required. This default setting is entirely sufficient for this calling direction. Nevertheless if a number manipulation or any other type of manipulation towards IPPBX is required, configure a mapping table and, if necessary, a complex function using the principle explained further below for the opposite direction (IPPBX towards OBS).</td>
</tr>
<tr>
<td></td>
<td>Via CLI:

<code>context cs SWITCH</code>
<code> routing-table called-e164 RT_FROM_OBS</code>
<code>   route default dest-interface IF_SIP_IPPBX</code></td>
</tr>
</table>

Routing Table from IPPBX to OBS

It is mandatory to configure the routing from IPPBX to OBS. Note that this part is highly dependent on the customer IPPBX / UC environment context.
The minimum mandatory configuration we strongly recommend here is the creation of the routing table from IPPBX towards OBS, plus the manipulation rules listed further below in this chapter.

Actions	Screenshot
Create routing table RT_FROM_IPPBX	Via Web UI: Open the menu Routing > Routing Table, then click on '+' to create a new call routing table. • Enter the name 'RT_FROM_IPPBX' and confirm with OK. • Under 'match' enter the matching type of the routing table, in this case 'called-e164'. (Same principle as for RT_FROM_OBS -> see previous chapters for screenshots)
Create a table entry in RT_FROM_IPPBX	In the newly created routing table, create a new routing table entry by clicking on the '+' button under the table list. In the next window select the following settings: • Match called-e164: select 'Default'. This means that the route will be chosen for any called e164 number (default call route in this table). • Call Destination: Type -> select 'dest-service'. This means that the destination of the route will be a service, more exactly the Hunt-Group service for OBS BT SIP-Trunks: HG_OBS_BTIP or HG_OBS_BTALK (see chapter Configure SIP-Trunk Hunt Group). If both SIP-Trunks have to be used from the same eSBC, you need to use dedicated prefixes in the previous step instead of default to separate the routing. • Name: select the correct hunt group: 'HG_OBS_BTIP' or 'HG_OBS_BTALK' facing the required SIP-Trunk. • Function to apply: select 'Complex Function' and choose CF_IPPBX_TO_OBS (see how to proceed at the end of chapter Mapping Table)

- Implement number format normalization towards OBS. For more details, see chapter Numbers Manipulations.
- Implement Calling Party Number translation / mapping from IPPBX to OBS, in order to translate internal / private to external / public numbers. See chapter Numbers Manipulations.
- Implement From Header manipulation in case of anonymous outgoing calls from IPPBX. For details, see chapter SIP Header manipulations / From, PAI/PPI headers for anonymous calls.

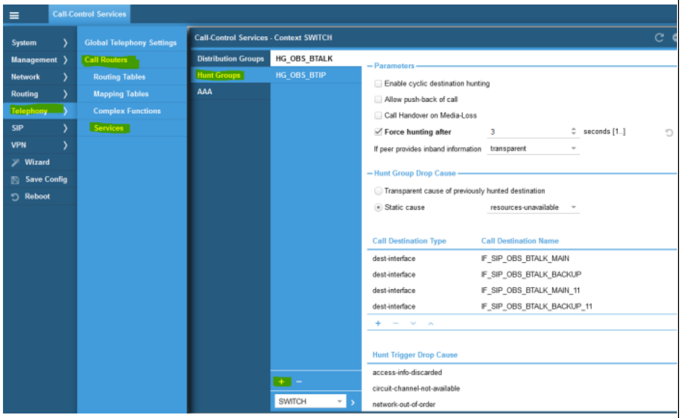
5.2.6 Configure SIP-Trunk Hunt Group

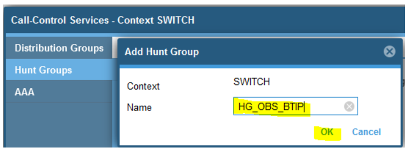
A Hunt-Group is an internal call routing service of Patton eSBC that provides redundancy for calls towards BTIP/BTalk SIP-Trunk. There are several destinations configured in a hunt-group. Those destinations can be SIP interfaces (this will be our case), routing tables or TDM interfaces in case of a hybrid eSBC.

A Hunt-Group accepts a call that is routed to it and sets up a second call that is placed to the first configured destination. If this destination is not reachable, another destination is tried until one of the configured destinations accepts the call.

It works in conjunction with penalty-box feature of the SIP Interfaces, which uses SIP OPTIONS keepalive, to automatically select working SIP peer without even trying to send traffic to a not-responding peer.

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td>1. Create Hunt-Group service for BTIP</td><td>
Via Web UI:

Open the menu Telephony > Call Routers > Services > Hunt Groups, then click on ‘+’ button to create a new hunt-group:


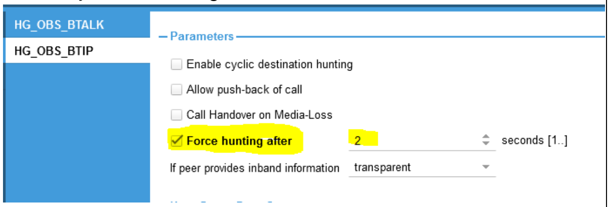
Enter the hunt-group name HG_OBS_BTIP and confirm with OK:


Select the created hunt-group HG_OBS_BTIP and modify only the following parameters. The other parameters must be left to the default values:

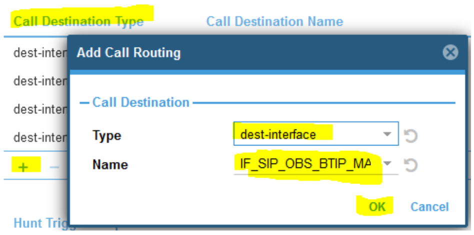
• Check the box ‘Force hunting after’ and set a duration value (in seconds) after which the hunting will be triggered on no response.
</td></tr>
</table>

Actions	Screenshot

We recommend a setting of 1s (or max 2s), which corresponds to the acceptable delay of SIP non-response after which the eSBC should try the next configured destination:



- Under ‘Call Destination Type’ create a destination type by clicking on ‘+’, then select the type ‘dest-interface’ in the drop-down list, then select the SIP interface name IF_SIP_OBS_BTIP_MAIN (created in the previous chapter Configure SIP Interfaces) and confirm with OK:

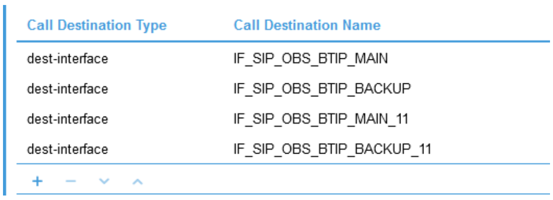


- Repeat the previous step by creating the same destination type with the SIP interface IF_SIP_OBS_BTIP_BACKUP

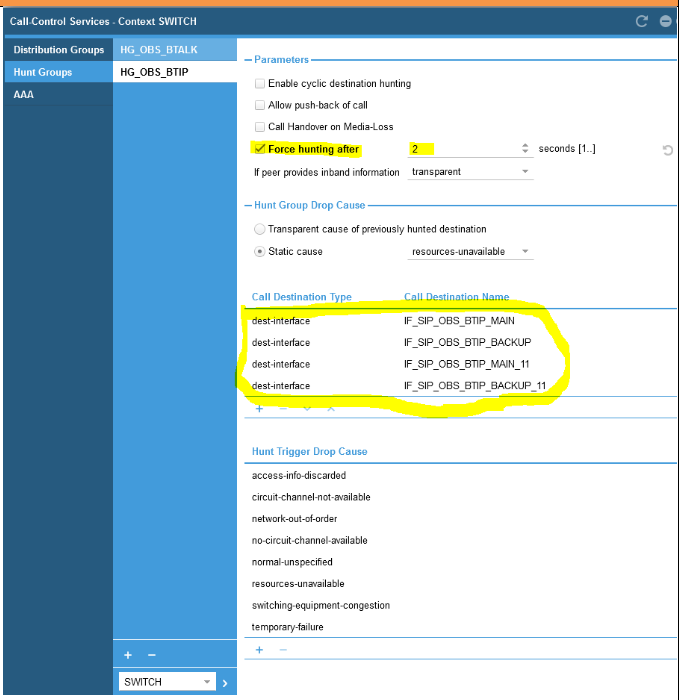
Optional (resiliency model):

- Repeat the previous step by creating the same destination types with the SIP interfaces:F_SIP_OBS_BTIP_MAIN_11 and IF_SIP_OBS_BTIP_BACKUP_11
(also created in the previous chapter Configure SIP Interfaces)

The destinations list must have this order after creation:



Actions	Screenshot
	If the order of creation is different, you can modify it by selecting a destination-interface and moving it to the right place using the arrow up and down buttons. It is important to follow this logic: 1. 1st destination = SIP Interface having as 'remote' the main IP@ of OBS SBC and as 'local' the main local IP@ 2. 2nd destination = SIP Interface having as 'remote' the backup IP@ of OBS SBC and as 'local' the main local IP@ Optional (resiliency model) 3. 3rd destination = SIP Interface having as 'remote' the main IP@ of OBS SBC and as 'local' the backup local IP@ 4. 4th destination = SIP Interface having as 'remote' the backup IP@ of OBS SBC and as 'local' the backup local IP@ • Hunt Group Drop Cause: The displayed list of drop causes is the default one and should not be changed. This is how the whole Hunt-Group must look like after all necessary settings for BTIP have been done, including the optional local resiliency:

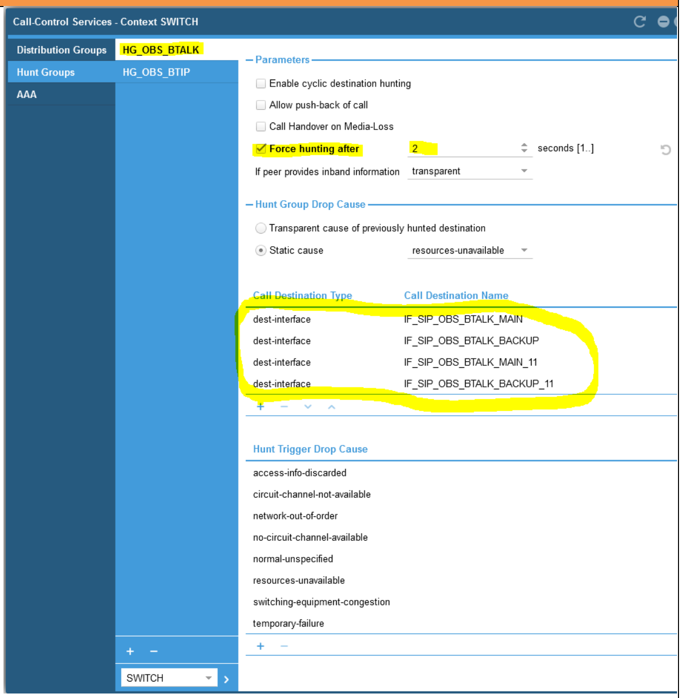
<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>

Via CLI

<code>service hunt-group HG_OBS_BTIP</code>
<code> timeout 2</code>
<code> route call 1 dest-interface IF_SIP_OBS_BTIP_MAIN</code>
<code> route call 2 dest-interface IF_SIP_OBS_BTIP_BACKUP</code>
<code> route call 3 dest-interface IF_SIP_OBS_BTIP_MAIN_11</code>
<code> route call 4 dest-interface IF_SIP_OBS_BTIP_BACKUP_11</code></td></tr>
<tr><td>2. Create Hunt-Group service for BTalk</td><td>Via Web UI:

Proceed exactly the same way as by creating Hunt-Group service for BTIP above, except that the destination SIP-Interfaces for BTalk must be selected.

This is how the whole Hunt-Group must look like after all necessary settings for BTIP have been done, including the optional local resiliency:</td></tr>
</table>

Actions	Screenshot
	

Via CLI:

```
service hunt-group HG_OBS_BTALK
  timeout 2
  route call 1 dest-interface IF_SIP_OBS_BTALK_MAIN
  route call 2 dest-interface IF_SIP_OBS_BTALK_BACKUP
  route call 3 dest-interface IF_SIP_OBS_BTALK_MAIN_11
  route call 4 dest-interface IF_SIP_OBS_BTALK_BACKUP_11
```

5.2.7 SIP Header Manipulation

For unencrypted or encrypted BTalk/BTIP SIP Trunk architecture, it is required to implement some Message Manipulation for the outgoing message toward Orange BTalk/BTIP. Those Manipulations Rules are detailed on the chapter SIP rules & manipulations (SBC Application). Please jump to this Chapter directly.

5.3 OBS Business Talk over Internet & BTIP over Internet Carrier North encrypted SIP configuration for Patton SBC (TLS)

As a prerequisite Patton recommends reading the Smartnode_SBC_Security_Guide to understand how to secure Patton eSBC in your network infrastructure.

Optionally, we recommend to configure ACL for WAN IP Interface locally in addition to the global internet firewall filtering -> see in annex eSBC local security ACL.

5.3.1 Configure a Certificate for the eSBC

Business Talk Over Internet & Business Talk IP Over Internet only allows TLS connections from the eSBC for SIP traffic with a certificate signed by one of the trusted public certification authorities.

To obtain this Certificate Authority (CA) you must generate your CSR based on the information of the SBC and Company with SHA-256 encryption.

The mentioned parameters in the table below are the one specific to Customer. It is just an example of CSR for a Company “Enterprise_test” located in Paris France with an SBC with FQDN name “SBC123.enterprise_test.com” resolving Public IP 83.206.61.113

Common Name	Organizational Unit	Company name	Locality or city name	Country code
SBC123.enterprise_test.com	-	COMPANY Enterprise	Paris	FR

1st Subject Alternative Name	2nd Subject Alternative Name	3rd Subject Alternative Name	Signature Algorithm	Private Key size
IP 83.206.61.113	_	_	SHA-256	2048

As soon as you receive the CA Root/Intermediate, you will have to load those on the Patton eSBC into the PKI folder and use them in the TLS Profile created for this interconnection with Orange BToI/BTIPoI.

Request a certificate for the eSBC External interface and its configuration is based on the following example:

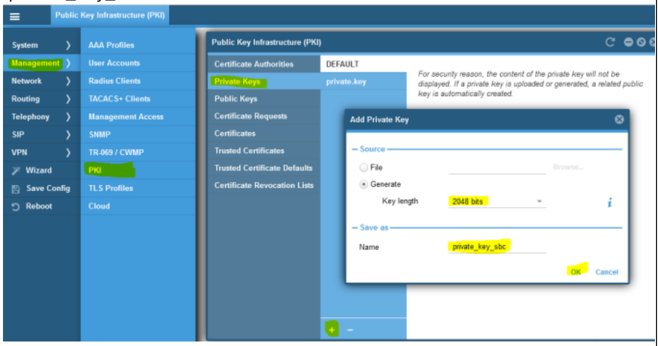
STEP 1: Generate a Certificate Signing Request (CSR) and obtain the certificate from a supported Certification Authority (CA

Generate a Certificate Signing Request (CSR) and obtain the certificate from a supported Certification Authority (CA)

Note:
The customer must ensure their eSBC FQDN's are resolved through a public DNS before generating the CSR

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td>1. Generate a private/public key pair on the device.</td><td>
Via Web UI:

Open the menu Management > PKI > Private Keys, then click on '+' button to generate a new private key.

Key length: leave the default value of 2048 bits
Name: provide an explicit name for the planned usage, for example 'private_key_sbc'. Click on OK to confirm.


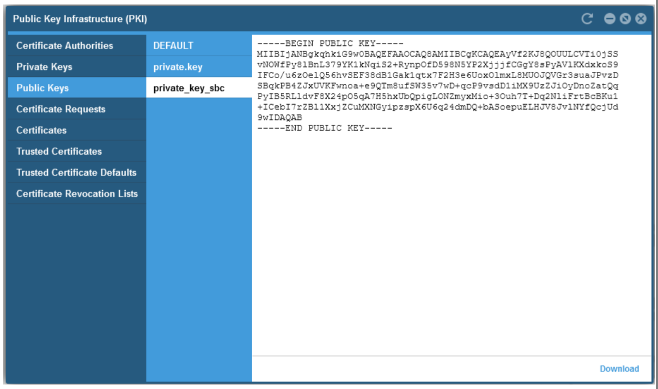
Via CLI:
<code>generate pki:private-key/private_key_sbc.key key-length 2048</code>

Note that it implicitly generates a public key as well.

The generated private key is only listed, without displaying its content for security reasons:
</td></tr>
</table>

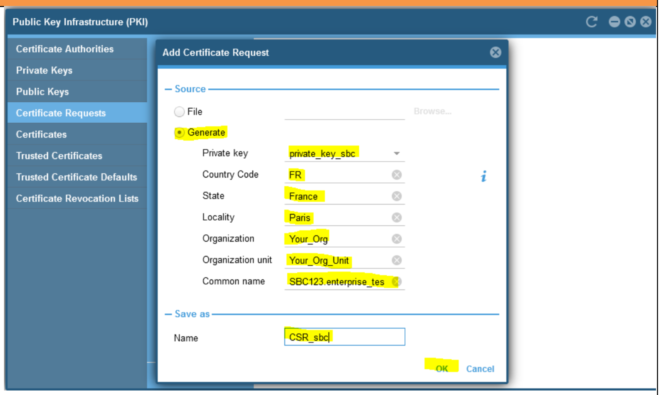
<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>


The generated public key can be seen under the submenu ‘Public Keys’ of the menu Management > PKI:

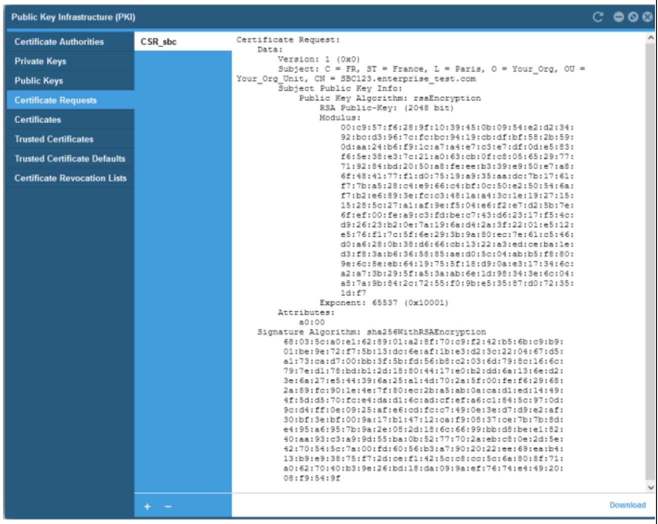


</td></tr>
<tr><td>2. Generate a Certificate Signing Request (CSR)</td><td>Under the submenu ‘Certificate Requests’, generate a CSR by setting the following parameters:

• Choose ‘Generate’
• Private key: select the previously created private key in the drop-down list
• Country code: use the two-letter ISO code for the country where your organization is located (example: FR for France)
• State: enter the state name (example: France)
• Locality: enter the locality name (example: Paris)
• Organization: enter the organization / company name
• Organization unit: enter the organization unit name
• Common name: enter the FQDN of the SBC, in our example SBC123.enterprise_test.com
• Save the CSR Request by entering a name, for example CSR_sbc.
• Click on OK to generate the CSR.</td></tr>
</table>

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>


This is how the CSR looks like after it has been generated:



Note that the most important parameter is the Common Name which MUST be equal to the defined FQDN of the SBC, which is resolved to its public IP address through DNS.

Via CLI:

generate pki:certificate-request/CSR_sbc private-key pki:private-key/private_key_sbc country FR state France locality Paris organization Your_Org organization-unit Your_Org_Unit common-name SBC123.enterprise_test.com
</td></tr>
</table>

Actions	Screenshot
3. Export CSR	Via Web UI: From the PKI submenu ‘Certificate Requests’ (see previous step) click on the previously created CSR and click on ‘Download’, then save the CSR on your computer. Via CLI: `export pki:certificate-request/CSR_sbc` Execution output example: *-----BEGIN CERTIFICATE REQUEST-----* *MIICpTCCAY0CAQAwYDELMAkGA1UEBhMCQ0gxDTALBgNVBAg* *NF9cuDx4qqsSIBIJ9Yv1C2X6T0WjTyOHQDICHAr58PTRT+MzR9* ... ... *y3f71W3oPz602akU48nRPPPrToFm4Z1zULiCrGGEhaMQK2bPMxoTt* *//HC/jCyNe+* *-----END CERTIFICATE REQUEST-----* Either copy the printout of the export command including the BEGIN / END headers from the terminal or use the following command to upload the request to a TFTP server: `#copy pki:certificate-request/CSR_sbc tftp://<server>/CSR_sbc`

When the CSR is generated copy the CSR text and send it to your Organization’s Certificate Authority (CA) which will sign it with its own private key and will return you the issued signed certificate in one of the usual file extension formats (*.cer, *.crt, .pem, .p12 etc), often included in a p7b bundle file. The Root and Intermediate Certificates (*.crt files) must be transmitted to Orange Business Services team.

When you get the CA files (p7b and bundle), please deploy them as explained bellow.

Make sure that the file is a plain-text file containing the "BEGIN CERTIFICATE" and “END CERTIFICATE" headers, as shown in the example of a Base64-Encoded X.509 Certificate below:

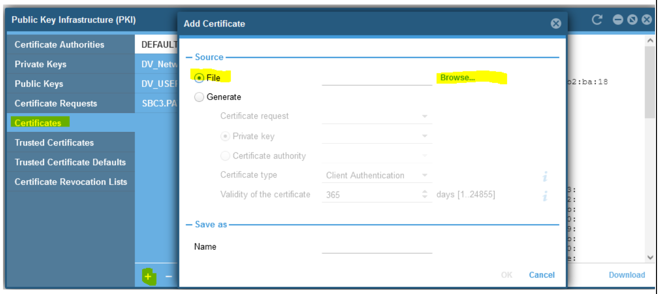
```
-----BEGIN CERTIFICATE-----

-----END CERTIFICATE-----
```

STEP 2: Deploy the SBC and Root/Intermediate Certificates on the SBC

After receiving the certificate from the certification authority, install the SBC Certificate and Root/Intermediate Certificates as follows

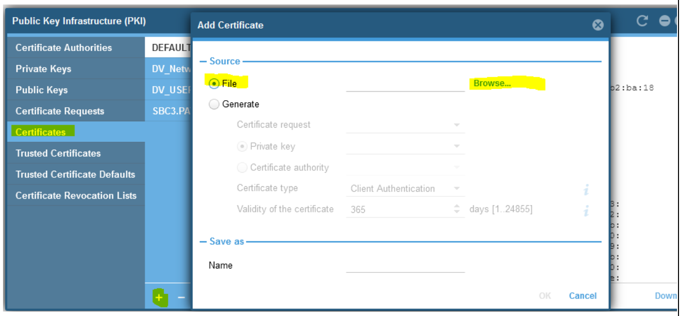
Note : Certificate supported formats are file with extension *.cer, *.crt, .pem, .p12)

SBC Certificate

Actions	Screenshot
1. Import the signed TLS Certificate received from the CA	Via Web UI: Open the menu Management > PKI > Certificates, then click on ‘+’ button to import a certificate, then select ‘File’ and browse to your TLS certificate received from your CA. Confirm with OK.  After this operation, you will be able to verify the data included in your TLS certificate by clicking on the certificate name. Verify if all the fields exactly correspond to the names you provided in the CSR, especially the Common Name which must be equal to the FQDN of the SBC. Via CLI: Copy the signed TLS certificate received from the CA from your TFTP server to the SBC, into the folder pki:certificate. In the example below, we considered a certificate file format *.crt, but you should adapt it to your file format: `copy tftp://<tftp_server>/SBC123.enterprise_test.com.crt pki:certificate/SBC123.enterprise_test.com.crt`

__Customer Root / Intermediate public Certificates__

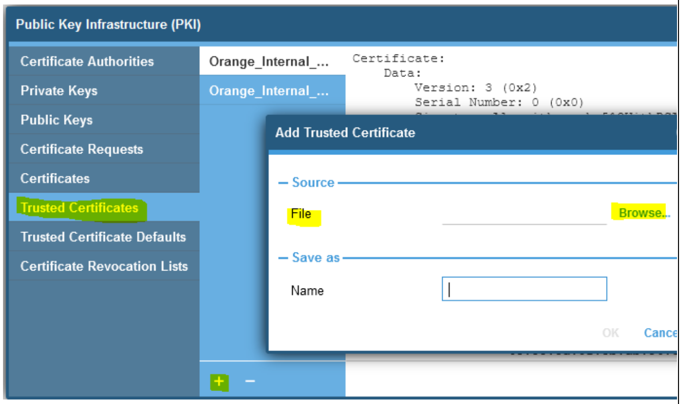
After receiving the certificate from the certification authority, install the SBC Certificate and Root/Intermediate Certificates as follows:

Actions	Screenshot
1. Deploy the public Root Certificate of your CA	Via Web UI: Use exactly the same process as for importing the SBC certificate: open the menu Management > PKI > Certificates, then click on '+' button to import a certificate, then select 'File' and browse to the file location of the Root certificate from your CA. Confirm with OK.  Via CLI: Copy the Root certificate of the CA from your TFTP server to the SBC, into the folder pki:certificate: `copy tftp://<tftp_server>/CA_ROOT.crt pki:certificate/CA_ROOT.crt`
2. Deploy the Intermediate Public Certificate of your CA	Via Web UI: repeat the same steps for the Intermediate certificate: Via CLI: proceed exactly the same way as in the previous step with CA_INTERMEDIATE.crt

STEP 3 : Communicate your Public CA Root and Intermediate Certificates authorities which signed your eSBC certificate to Orange BTALK project Team

STEP 4 : Import Orange Business Services Public Certificates Authorities

Ask Orange BTALK Team for the Orange Public CA Root and Intermediate Certificates which signed their infrastructure certificate, then import them on your Patton eSBC under Trusted Certificates.

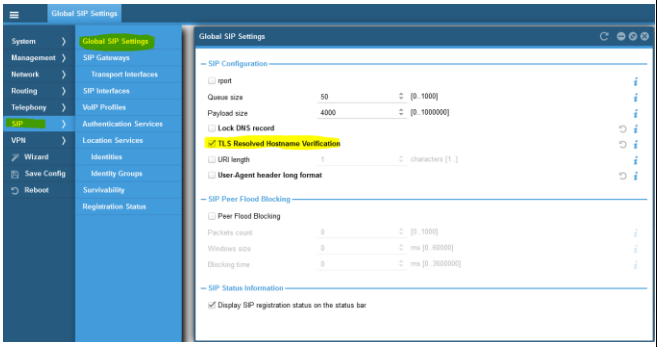
Actions	Screenshot
1. Import the Public Root Certificate of OBS	Via Web UI: Open the menu Management > PKI > Trusted Certificates, then click on ‘+’ button to import a certificate, then select ‘File’ and browse to the file location of the Root certificate from OBS. Confirm with OK.  Via CLI: Copy the Root certificate of OBS from your TFTP server to the SBC, into the folder pki:trusted-certificates: `copy tftp://<tftp_server>/CA_ROOT.crt pki:trusted-certificate/CA_ROOT.crt`
2. Import the Intermediate Public Certificate of OBS	Via Web UI: proceed exactly the same way as in the previous step. Via CLI: proceed exactly the same way as in the previous step.

5.3.2 Configure global SIP TLS settings

This part allows the configuration of some global SIP security relevant settings. More precisely we only need to activate one option: TLS resolved hostname verification.

This option allows SIP to match the subject alternate names or the common name of TLS certificates against domain names discovered through DNS instead of matching them against the configured source domain name only.

Additionally, since the SW version 3.20.4, it also allows to match the IP address from the Contact header of incoming from the SIP Trunk against already resolved domain names from local DNS cache, which is necessary in case of BTIPoI / BToI because it uses IP addresses instead of the hostname in the contact header. This parameter is very important and prevents potential issues in case of subsequent SIP requests from the eSBC on incoming calls, like sending out SIP UPDATE, re-INVITE or BYE messages in case of long duration incoming calls over BTIPoI / BToI. Note that enabling this option might be not compliant with all security requirements of RFC 6125.

Actions	Screenshot
1. Enable the Global SIP TLS Resolved Hostname Verification parameter	Via Web UI: Open the menu Management > SIP > Global SIP Settings and just check the box "TLS Resloved Hostname Verification".  Via CLI: `sip` `  tls resolved-hostname-verification`

5.3.3 Configure TLS Profile

The TLS profile defines the crypto parameters for the SIP protocol.

The encrypted architecture requires the usage of an encryption Key and Ciphers present in a TLS Context in order. A specific Orange BTALK TLS Context have to be created.

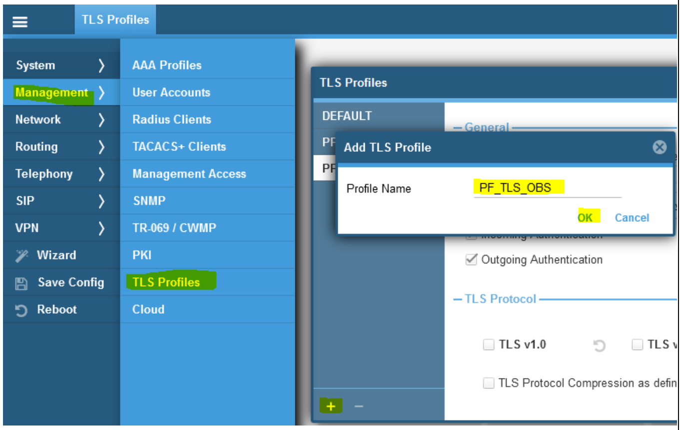
This SIP signaling will be configured to be compliant with Orange BTalk specifications:

- ✓ For encrypted BTALK/BTIP SIP Trunk architecture we need to configure TLS V1.2
- ✓ Key size 2048
- ✓ Cipher list is supported as Cipher Client/Server:
  - TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (Recommended)
  - TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
  - TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
  - TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
  - TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
  - TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
  - TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
  - TLS_DHE_RSA_WITH_AES_256_CBC_SHA256
- ✓ TLS Mutual authentication activated.

The mentioned parameters in the table below are the one specific to Orange Profile. All the other parameters must be left as «default value».

Parameter	Value
TLS Profile	TLS Orange
TLS protocol	TLS 1.2 Only
Mutual Authentication	Enabled
Client Cipher	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
Validate Server FQDN	Disabled
Client Certificate	<SBC SmartNode Certificate>
Validate Client FQDN	Disabled
Server Certificate	<SBC SmartNode Certificate>

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td>1. Create TLS Profile for BToI / BTIPoI</td><td>
Via Web UI:

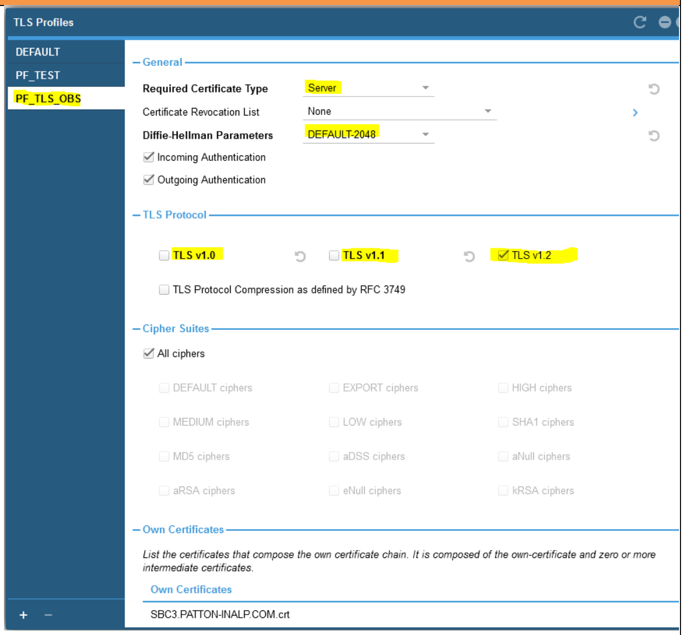
Open the menu Management / TLS Profiles, then click on ‘+’ to create a new TLS profile and name it accordingly, for example PF_TLS_OBS, and confirm with OK.


Via CLI: <code>profile tls PF_TLS_OBS</code>
</td></tr>
<tr><td>2. Configure the TLS Profile PF_TLS_OBS (part 1)</td><td>
Via Web UI:

In the Web submenu TLS Profiles, click on the newly created TLS profile and configure it the following way (only parameters that differ from the default settings are described):

Under General / Required Certificate Type, select ‘Server’
Under General / Diffie-Hellman Parameters, select ‘DEFAULT-2048’
Under TLS Protocol, disable (unselect) TLS v1.0 and TLS v1.1 and leave only TLS v1.2 enabled.

</td></tr>
</table>

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>


Via CLI:
<code>profile tls PF_TLS_OBS</code>
<code>no protocol tls-v1.0</code>
<code>no protocol tls-v1.1</code>
<code>diffie-hellman-parameters pki:diffie-hellman-parameters/DEFAULT-2048</code>
<code>require certificate-type server</code>
</td></tr>
<tr><td>3. Configure the TLS Profile PF_TLS_OBS (part 2)</td><td>
Via Web UI:

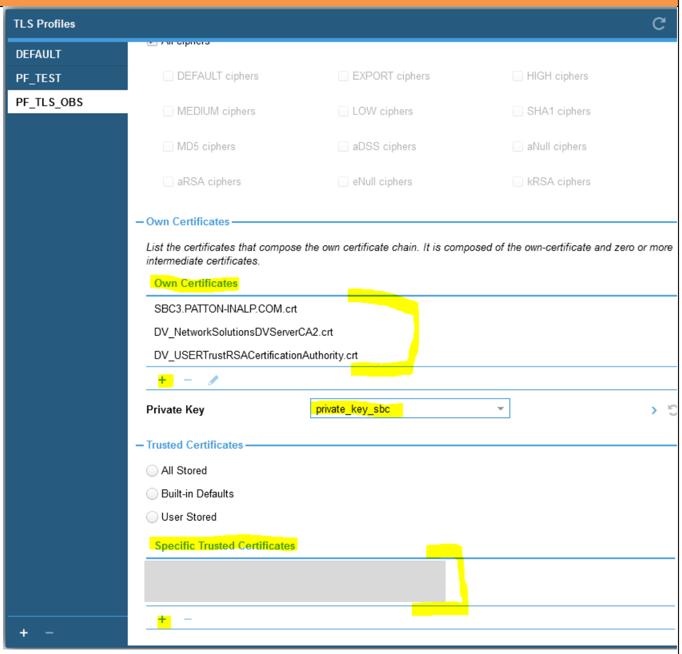
• Declare the previously imported SBC certificates:
1) the SBC own certificate (signed by the CA, containing the FQDN)
2) the Intermediate Certificate of the CA
3) the Root Certificate of the CA

For that task, just click on ‘+’ under ‘Own Certificates’ and select the SBC own certificate. Repeat the same step for the Intermediate and Root CA Certificate.
The examples from the screenshot below are form the certification test setup – use your own certificates in your setup.

• Under ‘Private Key’ select the previously created private key, which was used for the CSR.

• Declare the previously imported certificates from OBS under ‘Specific Trusted Certificates’ :
1) the Intermediate Certificate of OBS
2) the Root Certificate of OBS
</td></tr>
</table>

Actions	Screenshot



Via CLI:

```
profile tls PF_TLS_OBS
private-key pki:private-key/private_key_sbc
own-certificate 1 pki:certificate/SBC123.enterprise_test.com.crt
own-certificate 2 pki:certificate/CA_INTERMEDIATE.crt
own-certificate 3 pki:certificate/CA_ROOT.crt
trusted-certificate pki:trusted-certificate/
  <ORANGE_INTERMEDIATE_CA.pem>
trusted-certificate pki:trusted-certificate/
<ORANGE_ROOT_CA.pem>
```

5.3.4 Configure public network interface

In the TLS profile used for BToI / BTIPoI (SIP/TLS) the WAN interface is usually exposed to the public internet from the DMZ, so it is strongly recommended to use an Access Control List in order to restrict access, which is additionally explained in the chapter *Configure ACL for WAN IP Interface*.

Please see § Configure Network Interfaces (Context IP) for more details

5.3.5 Configure Location Service

The Location Service on Patton eSBC is used to define specific incoming or outgoing authentication credentials (if required), registration parameters or to additionally restrict incoming SIP requests to only certain domain names or SIP URI's (through regular expressions).

In our case we use the Location Service only to add 'user=phone' to the Request URI, From, To, PAI headers, especially in order to specify that the user-part of the URI should be interpreted as a telephone number (tel-URI).

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td>Create Location Service LS_OBS_TLS</td><td>Via Web UI:

Proceed exactly the same way as described in the Chapter BT / BTIP unencrypted SIP (UDP) / Configure Locations Service, by considering the following two differences:

• When you add the Location Service, set the following name: LS_OBS_TLS

• Under LS_OBS_TLS / DEFAULT / Call Outbound / Transport Protocol Preference, select 'forced' and select 'tls' in the drop-down list of transport protocols.

These are the only differences compared to SIP/UDP, all the other parameters must be configured exactly the same way as mentioned there.</td></tr>
<tr><td></td><td>Via CLI:

<code>location-service LS_OBS_TLS</code>
<code> match-any-domain</code>

<code> identity-group DEFAULT</code>

<code>   alias expression [0-9]+</code>
<code>   user phone</code>

<code>   authentication inbound</code>
<code>   authenticate none</code>

<code>   call outbound</code>
<code>   transport-protocol force tls</code></td></tr>
</table>

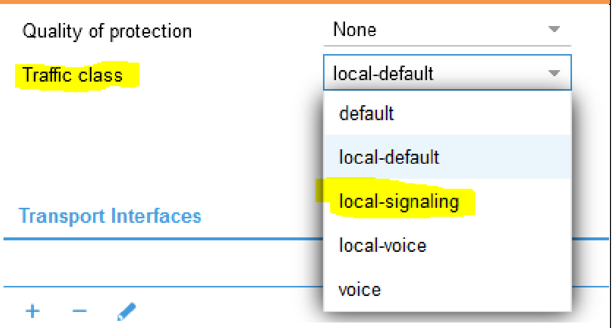
5.3.6 Configure SIP Gateway

The configuration of the SIP Gateway for SIP/TLS towards BToI / BTIPoI is nearly identical to the one described in the chapter SIP/UDP, except the following differences:

- Specific SIP Gateway name
- Only one SIP Gateway used towards BToI / BTIPoI, because for this scenario we do not implement Patton eSBC IP interface redundancy (only one local IP-address is in use instead of main + backup).

We omitted the screenshots for the Web User Interface configuration elements, because the menus have been shown in the SIP Gateway (UDP) configuration subchapter, in OBS BTIP unencrypted SIP chapter. Only the parameter values will be explained.

Actions	Screenshot
Access the SIP Gateway menu	Via Web UI: Open the **menu** SIP > SIP Gateways
Create the SIP Gateway GW_OBS_TLS	Click on '+' at the bottom left to create a new SIP Gateway, enter the name 'GW_OBS_TLS' and confirm with OK.
Select TLS Profile PF_TLS_OBS	Under TLS Profile, select the previously created TLS profile 'PF_TLS_OBS' from the drop-down list.
Enable TCP connection reuse	Check the option 'Enable TCP connection reuse', then also check the option 'As called party force TCP connection reuse' that appears below the previous one.
Select the correct traffic class for DSCP tagging for SIP signaling	Select the traffic class 'local-signaling'. Important: this traffic class has been configured in the DSCP profile (profile service-policy SP_WAN_OUT), as explained under Global configuration in the chapter DSCP profile. Select this setting in order to ensure the corresponding packet tagging of outgoing SIP messages towards the BTIP/BTalk SIP-Trunk. Leaving the default setting here would mean no DSCP tagging, so don't miss this part.

Actions	Screenshot
	
Create transport interface	As any other name of variable (by convention in capital letters), you are also free to define a name for the transport interface inside the SIP Gateway. It is through this interface that the binding with an IP address from the context IP / Interface is done. By convention we set the name IF_GW_OBS_TLS.
Edit the the created transport interface	Edit the settings of the newly created transport interface.
Modify the default settings inside the transport interface	Under binding select 'IP interface' and select the existing IP interface 'WAN_TLS' and select the IP address 'WAN_TLS_IP', both created previously in the Configure Network Interfaces (Context IP). Under 'Transport Protocol' check the box 'Enable TLS'. Leave the default port setting for TLS at 5061 or modify it if specified differently by OBS.
Bind the location service LS_OBS_TLS	Under 'Bound Location Service' click on '+' to bind a Location Service to the created SIP Gateway GW_OBS_TLS. Select the correct Location Service: Choose the previously created Location Service LS_OBS_TLS (described in previous chapter Configure Location Service) Leave the option „Disable registration outbound“ unselected (default). Despite this option, no outgoing registration registration will take place, because it is not activated in the selected Location Service.
Enable the SIP Gateway GW_OBS	Finally enable the SIP Gateway: under GW_OBS_TLS / Gateway State, check the box 'Enable'. Note that straight after enabling, the Smartnode starts communicating with the SIP-Trunk in both directions through this SIP Gateway.

	Via CLI: `context sip-gateway GW_OBS_TLS` `  use profile tls PF_TLS_OBS` `  bind location-service LS_OBS_TLS` `  traffic-class local-signaling` `  interface IF_GW_OBS_TLS` `    no transport-protocol udp+tcp` `    transport-protocol tls 5061` `    bind ipaddress ROUTER WAN_TLS WAN_TLS_IP` `context sip-gateway GW_OBS_TLS` `  connection-reuse forced` `  no shutdown`

5.3.7 Configure VoIP Profiles

Refer to chapter SIP / UDP -> Configure VoIP Profiles for introduction and general explanation about specified media codecs for BTIPoI / BToI. In this chapter we will apply the following codecs:

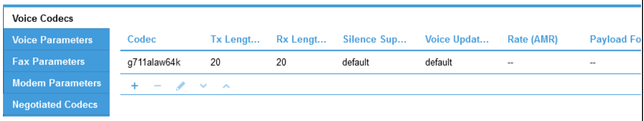
VoIP Profile for BTIPoI / BToI

- ***G.711 A-law 20 ms***
- ***G.711 µ-law 20ms (Optional)***

VoIP Codec Profile specific to Orange BTIPoI / BToI (Internet offer):

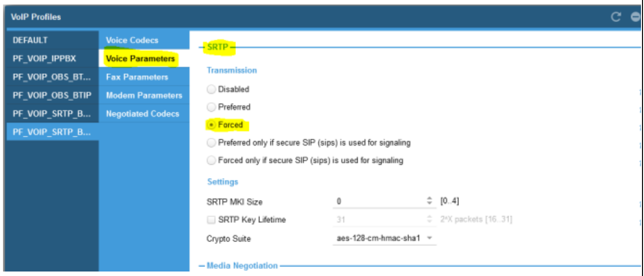
Actions	Screenshot
Create VoIP profile PF_VOIP_SRTP_BTIP	Via Web UI: Important remark: all settings except SRTP are identical to those of the VoIP profile for the unencrypted SIP-Trunk, so we just mention again the recommended configuration parameters. For details, please refer to the screenshots under BTIP unencrypted SIP Trunk -> VoIP Profile for BTIP. The only additional screenshot further below is related to SRTP / media encryption. Open the menu SIP > VoIP Profile , then click on '+' button to create a new VoIP profile and enter the name PF_VOIP_SRTP_BTIP (or PF_VOIP_SRTP_BTALK)
Proceed to audio codec modifications inside the profile	By default, a newly created VoIP profile has following two codecs defined in this order: 5. G.711 A-law 20 ms 6. G.711 µ-law 20 ms To remove G.711 µ-law from the codec list just select it and click on '-' button. Additionally, the arrow buttons under the list are used to modify the codec order after you have added all required codecs. Click on the codec G.711alaw64k, then on the pencil (edit) button to edit parameters of the codec. In the 'Edit Voice Codec' window disable Silence Suppression in the dropdown list (not supported by BTIPoI/BToI SIP-Trunks), which is enabled by default on each created codec. Leave the other parameters to the default values, i.e. Tx Length and Rx Length to 20ms and Voice Update Frames to default, which is disabled by default if Silence Suppression is disabled (Voice Update Frames can be effectively enabled only if Silence Suppression is enabled). Confirm the changes with OK.

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>Result:



</td></tr>
<tr><td>Modify Voice Parameters</td><td>Under the same VoIP profile, click on the next submenu ‘Voice Parameters’ and modify only these three parameters from their default values (and leave all the other parameters unchanged):

- DTMF Relay: check the box ‘enable’ and set the method to RTP, in order to use RTP payload for DTMF digits (RFC 2833/4733)
- SRTP: select ‘Forced’



- Crypto Suite: leave the default value AES-128-CM-HMAC-SHA1-80
- Media Negotiation:
    - Check the box ‘Announce ptime’ to add the attribute a=ptime:20 in the SDP of SIP messages sent by the eSBC.
    - Check the box ‘Response Single Codec’ to transmit only the negotiated codec in SDP of 200 OK responses instead of the codec list.
</td></tr>
<tr><td>Add T38 Fax relay</td><td>Under the same VoIP profile, click on the next submenu ‘Fax Parameters’ in order to add the T38 Fax relay capability.

Click on ‘+’ to add new fax transmission type.

In the ‘Add Fax Transmission’ window just leave the default selections ‘relay’ and ‘t38-udp’ and confirm:</td></tr>
</table>

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>Patton eSBCs transparently transmit the T38 fax relay in pass-through mode between the ATA device (with fax machine) and the SIP-Trunk, from one leg to the other, meaning that they are not able to transcode, for example between G711 and T38. Patton analog Gateways / ATAs in contrary are able to terminate T38.

Both Patton eSBCs and Gateways support Fax G3 standard over T38, with speeds of up to 14400 kbits/s and typically operate at 9600 bits/s. Super G3 can only be supported in conjunction with the bypass method with G.711 (see above). G.711 bypass for T38 should be disabled for OBS BTIP/BTalk. In this case only G3 with speeds up to 14400 kbits are supported, without Fallback capability.</td></tr>
<tr><td>Enable Codec Negotiation</td><td>Important: SIP protocol offers a codec negotiation mechanism. It is not guaranteed that the first codec in the SDP list will be used to set up the connection. Each codec in the list may be used.

On Patton eSBCs the codec negotiation is disabled by default in the VoIP profile, which honors the codec lists from each call leg independently, formed out of the remote and local capabilities. On HW DSP-based eSBC models, the DSP is inserted into the RTP path to make sure each side can use its codec. If necessary, the DSP will transcode between the codecs of the two RTP streams. Enabled “codec negotiation” will keep the DSP out of the picture in established calls and tries to negotiate a common codec for both call legs. We recommend to enable “codec negotiation” only on SN-models without HW DSP processors (SN500, vSN, SN5301 … see details in the list of the certified product versions)

Configurable only via CLI:

<code>profile voip PF_VOIP_OBS_BTIP</code>
<code> codec negotiation</code></td></tr>
<tr><td></td><td>Configuration method of the same complete VoIP Profile via CLI:

<code>profile voip PF_SRTP_OBS_BTIP</code>
<code> codec 1 g711alaw64k rx-length 20 tx-length 20</code>
<code> codec negotiation</code>   <--------- only on models without HW DSP
<code> dtmf-relay rtp</code>
<code> sdp-ptime-announcement</code>
<code> srtp transmission forced</code>   <--------- Important, enable SRTP
<code> codec response single</code>
<code> fax transmission 1 relay t38-udp</code></td></tr>
</table>

5.3.8 Configure SIP Interfaces

Two different configuration scenarios are to be considered here, according to the DNS Query method used:

1) Setup with DNS Query Type A (IPv4)

2) Setup with DNS SRV record

The two configurations differ from the concept point of view.

With DNS Query Type A, the two remote FQDNs (Nominal and Backup) need to be declared in dedicated SIP interfaces as remote peer. Additionally, we have one SIP interface per remote platform type (BToI and BTIPoI) due to the fact that the media / VoIP profile, which is called in SIP interface, differs. So in this scenario there will be 4 SIP Interfaces.

With DNS SRV Record Query, only one remote FQDNs needs to be declared in dedicated SIP interfaces as remote peer. Only 2 SIP Interfaces will be created: one per platform (BToI and BTIPoI).

Orange BToI / BTIPoI (SIP/TLS, SRTP) with DNS Type A

The configuration of SIP Interfaces on Patton eSBC for Orange BToI / BTIPoI is very similar to the one of SIP Interfaces for Orange BT / BTIP with several important differences that we will describe in this chapter.

All the detailed feature and configuration description of that chapter remains valid for this encrypted architecture. Only the relevant configuration parameter for the encrypted Orange SIP-Trunk will be described in detail in this chapter.

Patton eSBC will be configured to be compliant with Orange BToI / BTIPoI specification:

- ✓ For **encrypted BT SIP Trunk** architecture, we need to configure **TLS port 5061**
- ✓ For SIP-Trunk keep alive done with "**Options**" message (every 300 seconds)
- ✓ One SIP GW will be configured, as no local IP redundancy is used for this architecture

SIP Profile must be configured to be compliant with Orange BTalk/BTIP specifications:

- ✓ Session Timer is not supported

The mentioned parameters in the tables below are the one specific to Orange Profile. All the other parameters must be left as «default value».

SIP Interface	Host FQDN	Port	Protocol	VoIP Profile	Transport
1	<BT_Public FQDN_Nominal>	5061	TLS	Orange_BT_Profile or Orange_BTIP_Profile	Monitor: SIP Options Keep Alive Frequency: 300 Recovery frequency: 5
2	<BT_Public FQDN_Backup>	5061	TLS	Orange_BT_Profile or Orange_BTIP_Profile	Monitor: SIP Options Keep Alive Frequency: 300 Recovery frequency: 5

The SIP Interface parameters that are listed in the table below are only the non-default necessary parameters and values that shall be configured to respect Orange Certified Border specs.
All the other parameters of the SIP Interface configuration must be left at their respective default values.

Description	Parameter	Value	Comments
SIP call hold method to be used. Default setting: zero-IP, to be configured to the preferred method sendonly.	hold-method	direction-attribute sendonly	
Early Media handling according to RFC5009	early-media	accept authorized	
Do not accept incoming transferred calls from OBS with REFER method	call-transfer accept	no call-transfer accept	
Support REFER to re-INVITE conversion towards OBS	call-transfer emit	no call-transfer emit	REFER to Re-INVITE :When Blind and Consultative transfer are handled by the SIP REFER method, the SBC will generate a Re-INVITE towards the transfer target
Enable support of privacy and PAI/PPI headers	privacy	(just enable privacy)	
Apply the correct VoIP (media) profile	use profile voip	PF_VOIP_SRTP_BTIP (VoIP profile definition – see in corresponding chapter)	
OBS-specific header manipulation required in order to achieve the concatenated header content for User-Agent or Server header sent from eSBC (<IPBX Vendor v.X.X + SBC vendorV.X.X>**)**	use profile sip-tunneling out	OBS_USER_AGENT_CONCAT (*) (*): sip-tunneling profile definition - see in corresponding chapter	
Keep-alive OPTIONS	penalty-box sip-option-trigger	interval 300 timeout 300 force tls	
Session refresh method	session-timer	session-timer 1800 method update	

<u>Design concept used for the resilience on BT side:</u>

We consider the following inputs

- On OBS infrastructure side for encrypted SIP / TLS there is 2 SBCs pair (Nominal & Backup) for BToI / BTIPoI: in our example with FQDNs <BT_Public_FQDN_Nominal> & <BT_Public_FQDN_Backup> respectively
- On Patton eSBC each SIP Interface configuration contains by design a local and a remote host for proper signaling, that it will set into the host part of From header (local) and the host part of the To header (remote).

- Also each SIP interface calls a VoIP / media codec profile. There are two different VoIP profiles in use: PF_VOIP_SRTP_BTALK and PF_VOIP_SRTP_BTIP (defined in previous chapter)

This generates the following 4 combinations below with respective SIP logical Interface names used for resiliency purpose (Hunt group) :

	BTIP		(or BTalk)	
	Nominal SBC	Backup SBC	Nominal SBC	Backup SBC
Patton eSBC FQDN	IF_SIP_OBS_TLS_BTIP_MAIN	IF_SIP_OBS_TLS_BTIP_BACKUP	IF_SIP_OBS_TLS_BTALK_MAIN	IF_SIP_OBS_TLS_BTALK_BACKUP

The configuration of the 4 SIP Interfaces listed in this table are very similar. Only a few parameters differ: local / remote hosts and VoIP media profile.

In order to simplify the guidelines and not repeat the same description for all the SIP Interfaces, we will describe the configuration of all the 4 SIP Interfaces shown above only once by precising the different specific values that must be entered for each of them.

No screenshots are presented here. For screenshot details, refer to the SIP Interface chapter for unencrypted SIP configuration.

Actions	Screenshot
Create SIP Interface	Via Web UI: Open the menu SIP > SIP Interfaces, then click on '+' at the bottom left to create a new SIP Interface. Insert the SIP interface name according to remote platform (BToI / BTIPoI) and main / backup remote SBC: IF_SIP_OBS_TLS_BTIP_MAIN IF_SIP_OBS_TLS_BTIP_BACKUP or IF_SIP_OBS_TLS_BTALK_MAIN IF_SIP_OBS_TLS_BTALK_BACKUP
Configure basic settings of the SIP Interface	Select the submenu 'Basic Settings'. Select the following created variables and insert the following values: SIP Interface IF_SIP_OBS_TLS_BTIP_MAIN:

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>

Binding / SIP Gateway: choose the previously created SIP Gateway GW_OBS_TLS (see chapter SIP-Gateway towards BTIPoI / BToI SBC)
Call Destination: Type -> select ‘dest-table’ ; Name -> select RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX)
Profiles / VoIP: select the previously created VoIP profile PF_VOIP_SRTP_BTIP (see VoIP Profile for BTIPoI)
Remote: Host -> enter the FQDN name of the main SBC for BTIPoI / BToI <BT_Public FQDN_Nominal>; Port -> remote TCP listen port (5061)
Local: Host -> enter the local FQDN of the eSBC <eSBC_FQDN>; Port -> 5061. Important: eSBC FQDN and the Common Name of its TLS Certificate must match.

By proceeding the same way as for SIP Interface IF_SIP_OBS_TLS_BTIP_MAIN, select the other three SIP Interfaces listed in the previous table above and go through the same submenu ‘Basic settings’ by setting / entering the following values:

Interface IF_SIP_OBS_TLS_BTIP_BACKUP
Binding / SIP Gateway: GW_OBS_TLS
Call Destination: Type -> ‘dest-table’ ; Name -> RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX)
Profiles: VoIP -> PF_VOIP_SRTP_BTIP
Remote: Host -> BTIPoI/BToI Backup <BT_Public FQDN_Backup>; Port -> 5061
Local: Host -> eSBC FQDN: <eSBC_FQDN>; Port -> 5061

Or (for BTALK)

Interface IF_SIP_OBS_TLS_BTALK_MAIN
Binding / SIP Gateway: GW_OBS_TLS
Call Destination: Type -> ‘dest-table’ ; Name -> RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX)
Profiles: VoIP -> PF_VOIP_SRTP_BTALK
Remote: Host -> BTIPoI/BToI Backup <BT_Public FQDN_Nominal>; Port -> 5061
Local: Host -> eSBC FQDN: <eSBC_FQDN>; Port -> 5061
Interface IF_SIP_OBS_TLS_BTALK_BACKUP
Binding / SIP Gateway: GW_OBS_TLS
Call Destination: Type -> ‘dest-table’ ; Name -> RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX)

</td></tr>
</table>

Actions	Screenshot
	Profiles: VoIP -> PF_VOIP_SRTP_BTALK Remote: Host -> BTIPoI/BToI Backup <BT_Public FQDN_Backup>; Port -> 5061 Local: Host -> eSBC FQDN: <eSBC_FQDN>; Port -> 5061
Configure supplementary services of each SIP Interface	Select the submenu Supplementary Services in each SIP Interface. Uncheck the boxes Call Transfer Accept and Call Transfer Emit (which are enabled by default) in order to disable these methods: Proceed to the same modification on all SIP Interfaces towards OBS (see previous list).
Configure the SIP Features of each SIP Interface	Select the submenu SIP Features and modify the following parameters as described below on each SIP interface. • Enable Privacy and Asserted-Identity headers: (disabled by default) enable it in order to support sending Privacy and PAI/PPI headers towards the SIP-Trunk in appropriate call scenarios (typically for outgoing anonymous calls) according to RFC3323 and RFC3325. Note that some additional header manipulation is required in order for anonymous calls to work as specified for BTIP and BTalk -> see From, PAI/PPI headers for anonymous calls in the chapter SIP rules & manipulations (SBC Application). • Enable the session timer and configure it to 1800 seconds: the session refresh will be done each 1800 / 2 = 900 seconds (15 minutes). • As session timer method select 'update' in order to use the SIP method Update to refresh long duration calls. • Change the hold method from zero-ip (default) to direction-attribute-sendonly in order set the SDP attribute "sendonly" on Call Hold. • Enable the Penalty-Box feature: this feature checks the availability of the remote peer. • Enable the SIP Option trigger in order to activate the use of SIP Options Pings in correlation with the enables Penalty-Box feature. • Set the Interval and Timeout timers to 300 seconds. This is the time interval between two subsequent SIP Options messages sent by the eSBC through this SIP Interface. • Force the use of TLS transport protocol. We use this fix setting instead of the 'preferred' setting which combines UDP and TCP with a

Actions	Screenshot
	preference order, which is not necessary here because of the other interfaces dedicated to SIP/TLS/TCP. • Under Outgoing Calls Settings / URI-scheme, select SIP Change all those parameters the same way on the other seven SIP Interfaces towards OBS (see previous list).
Trusted hosts	Optional, useful for increased level of security at SIP level, additionally to the ACL lists already used on IP level. A list of trusted remote peers can be configured on SIP interfaces. If configured, only connections with peers in that list will be accepted. The list may contain IP-addresses or FQDNs. In case you would like to use this feature, select the check box 'Trust remote' and add the corresponding FQDN / IP-address of the remote peer.
Address Translation In	See chapter Diversion header – incoming calls
Address Translation Out	See chapters From, PAI/PPI headers for anonymous calls Diversion header – outgoing calls
Enable Early Media support according to RFC5009	Only via CLI: While the SIP dialog is in a provisional state (i.e., when the call is not connected yet), the P-Early-Media header defines with a direction attribute ("sendrecv", "sendonly", "recvonly", or "inactive") if early-media is allowed to be passed-through or if it has to be blocked by the SmartNode. With the new CLI command "early-media accept" the user can specify the early-media processing mode. The behavior of previous SW releases (prior to 3.20.1) is reflected by the option 'auto'. auto: No P-Early-Media header processing. Early media is accepted as soon as the device receives a provisional SIP response with SDP whose direction attribute allows the transmission. Further provisional SIP responses with SDP may change the current media direction whereas SIP responses without SDP have no effect on the current media direction. authorized: Early media is only accepted if explicitly authorized by the P-Early-Media header. Authorization happens with the P-Early-Media direction attribute ("sendrecv", "sendonly", "recvonly", or "inactive"), which can suppress a media direction that is enabled by SDP at the same time. Once a SIP response with SDP and with a P-Early-Media header has been received, further provisional responses with SDP may change the current

Actions	Screenshot
	media direction as long as they carry a P-Early-Media header as well, whereas SIP responses without SDP have no effect. OBS specification for BTIP / BTalk corresponds to the second option ‘authorized’, so following CLI is required: `early-media accept authorized`
Whole SIP Interface configuration via CLI	`context cs SWITCH` `interface sip <IF_SIP_TLS_BTIP>` `bind context sip-gateway <GW_OBS_TLS>` `route call dest-table RT_FROM_OBS` `remote <BT_Public FQDN_Nominal> 5061` `local <eSBC_FQDN> 5061` `hold-method direction-attribute sendonly` `early-media accept authorized` `no call-transfer accept` `no call-transfer emit` `privacy` `uri-scheme sip` `use profile voip <PF_VOIP_SRTP_BTALK> or voip <PF_VOIP_SRTP_BTIP>` `penalty-box sip-option-trigger interval 300 timeout 300 force tls` `session-timer 1800 method update`

Orange BTIPoI (SIP/TLS, SRTP) with DNS SRV

Considering the introduction description of this chapter, there are only 2 combinations here :

Patton eSBC	BTIPoI/BToI
Patton eSBC FQDN	BTIPoI or BToI FQDN
<eSBC_FQDN>	IF_SIP_TLS_BTIP

Actions	Screenshot
Create SIP Interface	Via Web UI: Open the menu SIP > SIP Interfaces, then click on ‘+’ at the bottom left to create a new SIP Interface. Insert the SIP interface name according to remote platform (BTIPoI): IF_SIP_OBS_TLS_BTIP
Configure basic settings of the SIP Interface	Select the submenu ‘Basic Settings’. Select the following created variables and insert the following values: SIP Interface IF_SIP_OBS_TLS_BTIP:

Actions	Screenshot
	• Binding / SIP Gateway: choose the previously created SIP Gateway GW_OBS_TLS (see chapter SIP-Gateway towards BTIPoI / BToI SBC) • Call Destination: Type -> select 'dest-table' ; Name -> select RT_FROM_OBS (see chapter Routing Table from OBS to IPPBX) • Profiles / VoIP: select the previously created VoIP profile PF_VOIP_SRTP_BTIP (see VoIP Profile for BTIPoI) • Remote: Host -> enter the FQDN name of the SBC for BTIPoI <BT_Public FQDN> (DNS SRV); Port -> do not enter any port ! This is important. Without remote port entry, the eSBC will trigger a DNS SRV Query Type, what we want here. If you enter a port number (5061), then a DNS Type A query will take place. • Local: Host -> enter the local FQDN of the eSBC <eSBC_FQDN>; Port -> 5061. Important: eSBC FQDN and the Common Name of its TLS Certificate must match.
Configure supplementary services of each SIP Interface	Select the submenu Supplementary Services in each SIP Interface. Uncheck the boxes Call Transfer Accept and Call Transfer Emit (which are enabled by default) in order to disable these methods: Proceed to the same modification on all SIP Interfaces towards OBS (see previous list).
Configure the SIP Features of each SIP Interface	Select the submenu SIP Features and modify the following parameters as described below on each SIP interface. • Enable Privacy and Asserted-Identity headers: (disabled by default) enable it in order to support sending Privacy and PAI/PPI headers towards the SIP-Trunk in appropriate call scenarios (typically for outgoing anonymous calls) according to RFC3323 and RFC3325. Note that some additional header manipulation is required in order for anonymous calls to work as specified for BTIP and BTalk -> see From, PAI/PPI headers for anonymous calls in the chapter SIP rules & manipulations (SBC Application). • Enable the session timer and configure it to 1800 seconds: the session refresh will be done each 1800 / 2 = 900 seconds (15 minutes).

Actions	Screenshot
	• As session timer method select 'update' in order to use the SIP method Update to refresh long duration calls. • Change the hold method from zero-ip (default) to direction-attribute-sendonly in order set the SDP attribute sendonly on Call Hold. • Enable the Penalty-Box feature: this feature checks the availability of the remote peer. • Enable the SIP Option trigger in order to activate the use of SIP Options Pings in correlation with the enables Penalty-Box feature. • Set the Interval and Timeout timers to 300 seconds. This is the time interval between two subsequent SIP Options messages sent by the eSBC through this SIP Interface. • Force the use of TLS transport protocol. We use this fix setting instead of the 'preferred' setting which combines UDP and TCP with a preference order, which is not necessary here because of the other interfaces dedicated to SIP/TLS/TCP. • Under Outgoing Calls Settings / URI-scheme, select SIP Change all those parameters the same way on the other seven SIP Interfaces towards OBS (see previous list).
Trusted hosts	Optional, useful for increased level of security, additionally to the ACL lists already used on IP level. A list of trusted remote peers can be configured on SIP interfaces. If configured, only connections with peers in that list will be accepted. The list may contain IP-addresses or FQDNs. In case you would like to use this feature, select the check box 'Trust remote' and add the corresponding FQDN / IP-address of the remote peer.
Address Translation In	See chapter Diversion header – incoming calls
Address Translation Out	See chapters From, PAI/PPI headers for anonymous calls Diversion header – outgoing calls
Enable Early Media support according to RFC5009	Only via CLI: While the SIP dialog is in a provisional state (i.e., when the call is not connected yet), the P-Early-Media header defines with a direction attribute

Actions	Screenshot
	("sendrecv", "sendonly", "recvonly", or "inactive") if early-media is allowed to be passed-through or if it has to be blocked by the SmartNode. With the new CLI command “early-media accept” the user can specify the early-media processing mode. The behavior of previous SW releases (prior to 3.20.1) is reflected by the option ‘auto’. auto: No P-Early-Media header processing. Early media is accepted as soon as the device receives a provisional SIP response with SDP whose direction attribute allows the transmission. Further provisional SIP responses with SDP may change the current media direction whereas SIP responses without SDP have no effect on the current media direction. authorized: Early media is only accepted if explicitly authorized by the P-Early-Media header. Authorization happens with the P-Early-Media direction attribute ("sendrecv", "sendonly", "recvonly", or "inactive"), which can suppress a media direction that is enabled by SDP at the same time. Once a SIP response with SDP and with a P-Early-Media header has been received, further provisional responses with SDP may change the current media direction as long as they carry a P-Early-Media header as well, whereas SIP responses without SDP have no effect. OBS specification for BTIP / BTalk corresponds to the second option ‘authorized’, so following CLI is required: `early-media accept authorized`
Whole SIP Interface configuration via CLI	`context cs SWITCH` `interface sip <IF_SIP_TLS_BTIP>` `bind context sip-gateway <GW_OBS_TLS>` `route call dest-table RT_FROM_OBS` `remote <BT_Public_FQDN_Nominal>` `local <eSBC_FQDN> 5061` `hold-method direction-attribute sendonly` `early-media accept authorized` `no call-transfer accept` `no call-transfer emit` `privacy` `uri-scheme sip` `use profile voip <PF_VOIP_SRTP_BTALK> or voip <PF_VOIP_SRTP_BTIP>` `penalty-box sip-option-trigger interval 300 timeout 300 force tls` `session-timer 1800 method update`

IPPBX

This part is configurable only on eSBC models with HW DSP (see Business Talk & BTIP Patton SmartNode eSBC certified versions).

We mention here only the parameter, which is relevant for the local ring-back tone generation towards IPPBX, when the provisional 180 Ringing response from BT/BTIP SIP-Trunk is either without SDP or with SDP and without P-Early-Media header, according to RFC3960, RFC5009 and the technical specifications for OBS BTIP / BTalk.

Actions	Screenshot
Enable local RBT generation towards IPPBX	Only via CLI: `interface sip IF_SIP_IPPBX` `  early-media emit forced`

5.3.9 Configure Call Routing

The Call Routing concept of Patton eSBS is explained in detail in the Call Routing chapter for OBS unencrypted SIP-Trunk.
In this chapter, only specificities related to the TLS configuration will be explained.

Routing Table from OBS to IPPBX

Incoming calls from OBS are received through one of the SIP Interfaces facing BT encrypted SIP-Trunk (see Configure SIP Interfaces). We will name it RT_FROM_OBS according the suggested naming convention.

Actions	Screenshot
Create routing table RT_FROM_OBS	Via Web UI: Open the menu Routing > Routing Table, then click on '+' to create a new call routing table. • Enter the name 'RT_FROM_OBS' and confirm with OK. • Under 'match' enter the matching type of the routing table, in this case 'called-e164'.
Create a table entry in RT_FROM_OBS	In the newly created routing table, create a new routing table entry by clicking on the '+' button under the table list. In the next window select the following settings: • Match called-e164: select 'Default'. This means that the route will be chosen for any called e164 number (default call route in this table). • Call Destination: Type -> select 'dest-interface'. This means that the destination of the route will be an interface. In our scenario (pure IP-IP eSBC with SIP and no TDM interfaces) this will be a previously configured SIP Interface. On hybrid eSBCs this could also be an ISDN or analog interface. • Name: select the SIP Interface 'IF_SIP_IPPBX' facing the IPPBX on the LAN side. • Function to apply: leave 'None' if no number manipulation is required. This default setting is entirely sufficient for this calling direction. Nevertheless if a number manipulation or any other type of manipulation towards IPPBX is required, configure a mapping table and, if necessary, a complex function using the principle explained further below for the opposite direction (IPPBX towards OBS).

Actions	Screenshot
	Via CLI: `context cs SWITCH` `  routing-table called-e164 RT_FROM_OBS` `    route default dest-interface IF_SIP_IPPBX`

Routing Table from IPPBX to OBS

It is mandatory to configure the routing from IPPBX to OBS. Note that this part is highly dependent on the customer IPPBX / UC environment context.
The minimum mandatory configuration we strongly recommend here is the creation of the routing table from IPPBX towards OBS, plus the manipulation rules listed further below in this chapter.

Actions	Screenshot
Create routing table RT_FROM_IPPBX	Via Web UI: Open the menu Routing > Routing Table, then click on '+' to create a new call routing table. • Enter the name 'RT_FROM_IPPBX' and confirm with OK. • Under 'match' enter the matching type of the routing table, in this case 'called-e164'. (Same principle as for RT_FROM_OBS -> see previous chapters for screenshots)
Create a table entry in RT_FROM_IPPBX	In the newly created routing table, create a new routing table entry by clicking on the '+' button under the table list. In the next window select the following settings: • Match called-e164: select 'Default'. This means that the route will be chosen for any called e164 number (default call route in this table). • Call Destination: 2 different configuration options, depending on the DNS Query type used: o Setup with DNS Query Type A (IPv4) Type -> select 'dest-service'. This means that the destination of the route will be a service, more exactly the Hunt-Group service for OBS BT SIP-Trunks, because the eSBC will hunt between the nominal and backup remote FQDN: HG_OBS_TLS_BTIP or HG_OBS_TLS_BTALK (see chapter Configure SIP-Trunk Hunt Group). Name: select the correct hunt group:

Actions	Screenshot
	HG_OBS_TLS_BTIP or HG_OBS_TLS_BTALK facing the required SIP-Trunk o Setup with DNS SRV record Type -> select 'dest-interface'. This means that the destination of the route will be directly a SIP interface, more exactly SIP interface for OBS BTIPoI / BToI SIP-Trunks: IF_SIP_OBS_TLS_BTIP (see chapter Configure SIP Interfaces). • Function to apply: select 'Complex Function' and choose CF_IPPBX_TO_OBS (see how to proceed at the end of chapter Mapping Table)

- Implement number format normalization towards OBS. For more details, see chapter Numbers Manipulations.
- Implement Calling Party Number translation / mapping from IPPBX to OBS, in order to translate internal / private to external / public numbers. See chapter Numbers Manipulations.
- Implement From Header manipulation in case of anonymous outgoing calls from IPPBX. For details, see chapter SIP Header manipulations / From, PAI/PPI headers for anonymous calls.

5.3.10 Configure SIP trunk Hunt Group

Only to consider for the DNS Query Type A scenario.

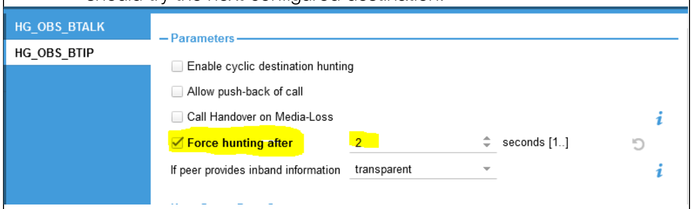
In case of use of DNS SRV Record, please ignore this chapter. (see explanation in the introduction of chapter SIP Interfaces)

A Hunt-Group is an internal call routing service of Patton eSBC that provides redundancy for calls towards BTIP/BTalk SIP-Trunk. There are several destinations configured in a hunt-group. Those destinations can be SIP interfaces (this will be our case), routing tables or TDM interfaces in case of a hybrid eSBC.

A Hunt-Group accepts a call that is routed to it and sets up a second call that is placed to the first configured destination. If this destination is not reachable, another destination is tried until one of the configured destinations accepts the call.

It works in conjunction with penalty-box feature of the SIP Interfaces, which uses SIP OPTIONS keepalive, to automatically select working SIP peer without even trying to send traffic to a not-responding peer.

In the table below we provide the configuration procedure without screenshots. For details with screenshots, please refer to the Hunt Group subchapter for the unencrypted BT SIP-Trunk.

Actions	Screenshot
Create Hunt-Group service for BTIPol	Via Web UI: Open the menu Telephony > Call Routers > Services > Hunt Groups, then click on '+' button to create a new hunt-group. Enter the hunt-group name HG_OBS_TLS_BTIP and confirm with OK. Select the created hunt-group HG_OBS_TLS_BTIP and modify only the following parameters. The other parameters must be left to the default values: • Check the box 'Force hunting after' and set a duration value (in seconds) after which the hunting will be triggered on no response. We recommend a setting of 1s (or max 2s), which corresponds to the acceptable delay of SIP non-response after which the eSBC should try the next configured destination:  • Under 'Call Destination Type' create a destination type by clicking on '+', then select the type 'dest-interface' in the drop-down list, then select the SIP interface name IF_SIP_OBS_TLS_BTIP_MAIN (created

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>in the previous chapter Configure SIP Interfaces) and confirm with OK:

• Repeat the previous step by creating the same destination type with the SIP interface IF_SIP_OBS_TLS_BTIP_BACKUP

The destinations list must have this order after creation:

Dest-interface: IF_SIP_OBS_TLS_BTIP_MAIN
Dest-interface: IF_SIP_OBS_TLS_BTIP_BACKUP

If the order is different, you can modify it by selecting a destination-interface and moving it up or down using the arrow buttons. It is important to follow this logic:

1. 1st destination = SIP Interface having as ‘remote’ the FQDN of the nominal OBS SBC for encrypted SIP-Trunk

2. 2nd destination = SIP Interface having as ‘remote’ the FQDN of the backup OBS SBC for encrypted SIP-Trunk

• Hunt Group Drop Cause:

The displayed list of drop causes is the default one and should not be changed.

Via CLI

<pre>service hunt-group HG_OBS_TLS_BTIP
  timeout 2
  route call 1 dest-interface IF_SIP_OBS_TLS_BTIP_MAIN
  route call 2 dest-interface IF_SIP_OBS_TLS_BTIP_BACKUP</pre></td></tr>
<tr><td>Or for BToI

Create Hunt-Group service for BToI</td><td>Via Web UI:

Proceed exactly the same way as by creating Hunt-Group service for BTIPoI above, except that the destination SIP-Interfaces must be selected from those for BToI.

Force hunting after: 2 seconds
Dest-interface: IF_SIP_OBS_TLS_BTALK_MAIN
Dest-interface: IF_SIP_OBS_TLS_BTALK_BACKUP

Via CLI:

<pre>service hunt-group HG_OBS_TLS_BTALK
  timeout 2
  route call 1 dest-interface IF_SIP_OBS_TLS_BTALK_MAIN
  route call 2 dest-interface IF_SIP_OBS_TLS_BTALK_BACKUP</pre></td></tr>
</table>

5.3.11 SIP Header Manipulation

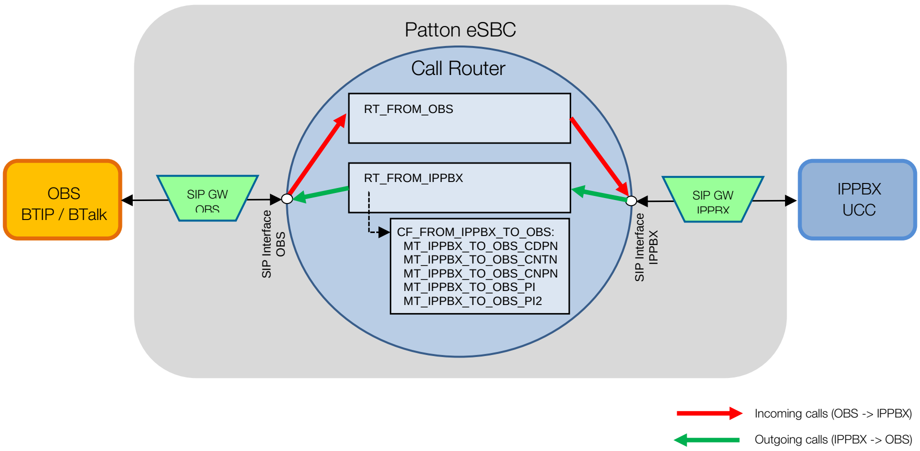
For unencrypted or encrypted BTalk/BTIP SIP Trunk architecture, it is required to implement some Message Manipulation for the outgoing message toward Orange BTalk/BTIP. Those Manipulations Rules are detailed on the chapter SIP rules & manipulations (SBC Application). Please jump to this Chapter directly.

5.4 SIP rules & manipulations (SBC Application)

This section provides the configuration regarding the device's SBC application, which is used for message rules & manipulations as described below.
This chapter is common to Orange BTalk ASBC encrypted or unencrypted BT SIP Trunk architecture.

5.4.1 Preamble: Manipulation principle on Patton eSBC

The following figure summarizes the Call Routing configuration for BT SIP Trunk on Patton eSBC with the focus on the main internal elements that implement SIP manipulations:



SIP Manipulation levels

Configuration Element	Available manipulation method	Types Headers Directions	Description
SIP Interface	Address Translation	Incoming	In incoming direction Address Translations allow to modify internal call properties of the Call Router of the SBC (like Called E164, Called Name, Called URI, Calling E164, Calling Name, Calling Redirection, Calling URI) using as input the SIP headers or the Request URI of the incoming call or a fix value set manually.
		Outgoing	In outgoing direction Address Translations allow to modify outgoing SIP headers or URIs (Contact header, Diversion header, From header, To header, Identity header, Request URI) using as input internal call properties of the Call Router.

	SIP Tunneling	Incoming	The SIP-tunneling profile allows tunneling of SIP headers from one SIP interface to another. Each profile defines a set of SIP headers and a set of SIP messages where the tunneling should be active. It is mainly intended to handle X-headers by tunneling them transparently or rename a standard header to an X-Header. This function is disabled by default. In our case we will use a specifically developed option of SIP-tunneling for the needs of User-Agent & Server header for OBS BTIP / BTalk.
		Outgoing	
Routing Table	Mapping Tables & Complex Functions (*)	Called E164	Translating of called E164 number
		Calling E164	Translating of calling E164 number
		IP Address	Translating of IP Address
		SIP URI	SIP URI Translation
		Type of Number	Translation of the type of number
		(many others…)	See CLI Reference Guide for more details.
SIP Gateway	IP@ / FQDN Spoofing (only outgoing)	Contact Header	Can be configured to another value manually
		Via Header	Can be configured to another value manually
		NAT Address	Can be configured to another IP@ manually

(*) If a Routing Table requires more than one **mapping table**, it makes call to a **Complex Function** which executes several configured Mapping Tables as displayed in the figure above

Therefore, several manipulation rules may be applied end-to-end. This is how the whole chain can look like. Note that the mentioned manipulations are optional and can be set or omitted at any level (by default they are not applied):
1) incoming address translations + SIP tunneling on the ingress SIP Interface
2) mapping tables in the Call Router
3) outgoing address translations + SIP tunneling on the egress SIP Interface
4) spoofing of Contact-, Via- headers and/or NAT address on the egress SIP-Gateway

5.4.2 SIP Messages Manipulations

No specific SIP Messages Manipulation is not necessary, because all the necessary handling respecting OBS requirements are already covered in chapters § 2.5.4 & 2.6.7 Configure SIP Interfaces

Additionally, SIP Header Manipulation", "Outbound Manipulations" and "Inbound Manipulations" are necessary, please § bellow

Commenté [BR5]: This chapter is not necessary, because all the necessary handling is already covered in chapters "SIP Header Manipulation", "Outbound Manipulations" and "Inbound Manipulations".

5.4.3 SIP Header manipulations

OBS-specific User-Agent and Server headers

The specific User-Agent and Server header content required by OBS (<IPBX Vendor v.X.X + SBC vendorV.X.X>) can be achieved by Patton eSBC through the SIP tunneling profile configuration element.

Object	Parameter	Value	Result
SIP Tunneling profile	Header	User-Agent	Concatenates User-Agent headers of IPPBX + eSBC to build a merged User-Agent towards OBS, when the profile is applied in Incoming direction on IPPBX side and in Outgoing direction on OBS side. The same is valid for the Server header.
SIP Tunneling profile	Header	Server	

The following SIP tunneling profile must be first created, then applied to the SIP Interface for IPPBX in incoming direction and to the SIP Interface for OBS in outgoing direction.

Actions	Screenshot
Create the SIP tunneling profile OBS_USER_AGENT_CONCAT	Only via CLI: `profile sip-tunneling OBS_USER_AGENT_CONCAT` `  header User-Agent` `  header Server`
Apply SIP tunneling profile to the SIP Interface facing the IPPBX in Incoming direction	`context cs` `  interface sip <IF_SIP_IPPBX>` `    use profile sip-tunneling in OBS_USER_AGENT_CONCAT`
Apply SIP tunneling profile to all SIP Interfaces towards BTIP / BTalk / BTIPoI / BToI in Outgoing direction	`context cs` `  interface sip <IF_SIP…>` `    use profile sip-tunneling out OBS_USER_AGENT_CONCAT`

From, PAI/PPI headers for anonymous calls

From Header manipulation is required in case of anonymous outgoing calls from IPPBX. These are calls with CLIR feature (calling line identification restriction) enabled.
The current SW implementation of Patton eSBC correctly handles the Privacy header (set to id) and P-Asserted-Identity / P-Preferred-Identity headers, but still sends the calling party identity and the actual domain name in the From header, which doesn't completely fulfill the technical specifications.
The specifications stipulate:

- set privacy header to id
- From header containing “anonymous” sip:anonymous@anonymous.invalid
- P-A-I containing the Calling party identification.

In order to set the correct user part in From, the following header manipulation is necessary on Mapping Table and SIP Interface levels:

Mapping Table level

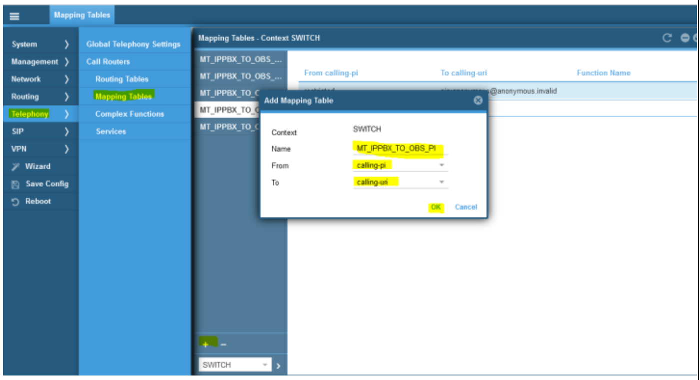
Following mappings must be applied to the call routing table from IPPBX to OBS (see figure in the preamble):

MT order	Mapping-Table name	Input Type of MT	Output Type of MT	Input Value	Output Value
1	MT_IPPBX_TO_OBS_PI	calling-pi	calling-uri	restricted	sip:anonymous@anonymous.invalid
2	MT_IPPBX_TO_OBS_PI2	calling-pi	calling-name	restricted	Anonymous

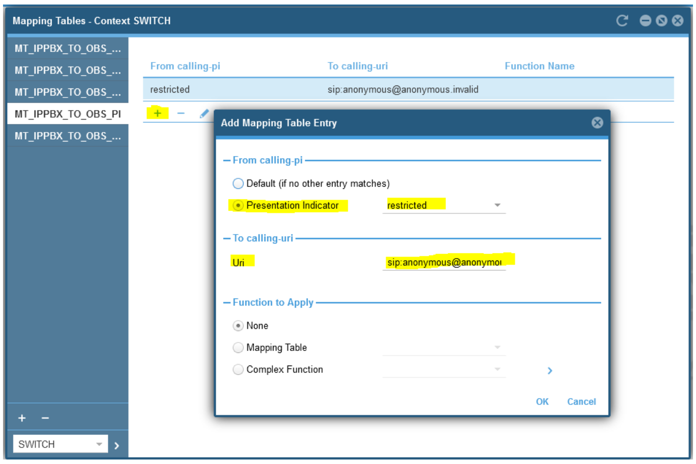
Explanation:
1: The first mapping transforms the Calling URI from <user@domain.com> to sip: anonymous@anonymous.invalid
2: As the rule 1 leaves the user display name unchanged, this 2nd rule overwrites also the display name with Anonymous in order to anonymize the name as well.

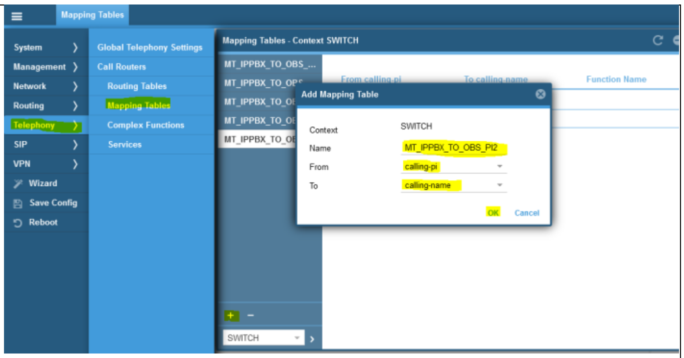
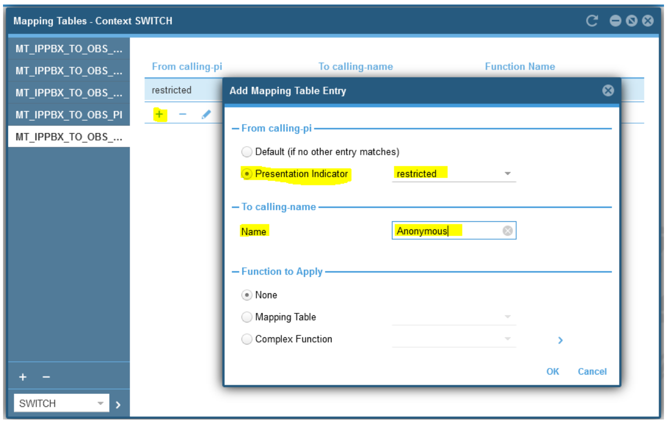
<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td>Configure Mapping table MT_IPPBX_TO_OBS_PI</td><td>Via Web UI:

Under Telephony > Mapping Tables, click on ‘+’ to create a new Mapping Table.
Name: MT_IPPBX_TO_OBS_PI
From: select ‘calling-pi’
To: select ‘calling-uri’
Confirm with OK



Select the created Mapping Table and create a new mapping table entry:</td></tr>
</table>

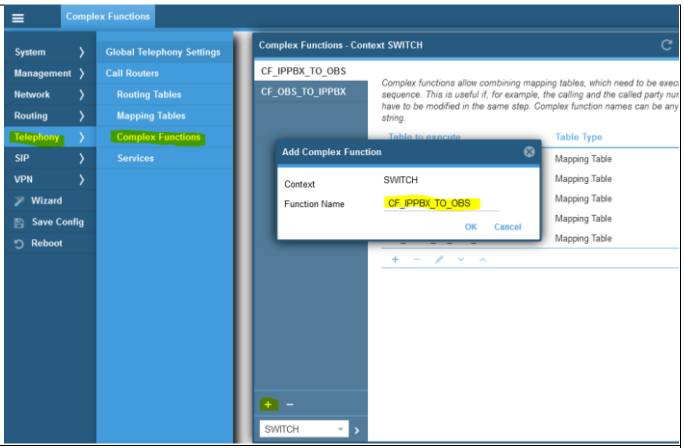
Actions	Screenshot
	Presentation indicator: select ‘restricted’ URI: enter ‘sip:anonymous@anonymous.invalid’ Confirm with OK
Configure Mapping table MT_IPPBX_TO_OBS_PI2:	Via Web UI: Under Telephony > Mapping Tables, click on ‘+’ to create a new Mapping Table. Name: MT_IPPBX_TO_OBS_PI2 From: select ‘calling-pi’ To: select ‘calling-name’ Confirm with OK 

<table>
<tr>
<td>Select the created Mapping Table and create a new mapping table entry:</td>
<td>
</td>
</tr>
<tr>
<td>Apply parameters :
Presentation indicator: select ‘restricted’
Name: enter ‘Anonymous’
Confirm with OK</td>
<td>

Via CLI:

<pre>
mapping-table calling-pi to calling-uri MT_IPPBX_TO_OBS_PI
  map restricted to sip:anonymous@anonymous.invalid

mapping-table calling-pi to calling-name MT_IPPBX_TO_OBS_PI2
  map restricted to Anonymous
</pre></td>
</tr>
</table>

Then create the complex function CF_FROM_IPPBX_TO_OBS as displayed	
Select the created complex function CF_FROM_IPPBX_TO_OBS	click on '+' button to add tables, then select MT_IPPBX_TO_OBS_PI in the mapping-tables drop-down list and confirm with OK. Repeat the same operation by adding MT_IPPBX_TO_OBS_PI2.

The next step with the SIP Interface is also required for the proper handling of anonymous calls.

SIP Interface level

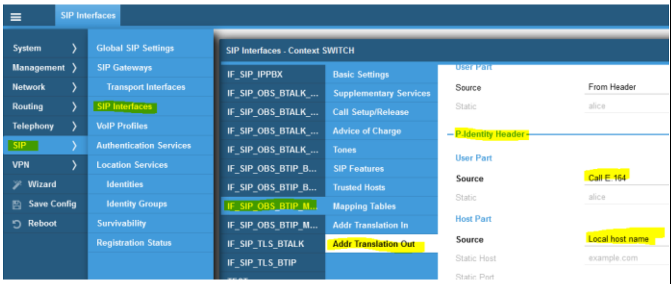
Following address translation must be applied to the SIP Interface towards OBS additionally to the previous step with the mapping table. It will modify the user part of PAI header in case of anonymous outgoing calls.

SIP Interface	Parameter	Value
<sip_interface_to_OBS>	address-translation outgoing-call identity-header user-part	call e164 single-user host-part local

Explanation:
Without this address translation, and after previous mappings in the call router, the outgoing PAI header towards OBS would contain <sip:anonymous@anonymous.invalid>.
This address translation transforms it to the originating PAI header received from the IPPBX in the form <calling_user@domain.com>, because the PAI header should not be anonymized.

On all SIP Interfaces configured towards BTIP, BTalk, BTIPoI, BToI proceed to the following configuration: go to the Web UI menu SIP > SIP Interfaces, choose one of the SIP Interfaces configured towards BTIP, BTalk, BTIPoI, BToI, then select the submenu 'Addr. translation Out' and under the configuration part 'P-Identity Header' select the following settings:

Actions	Screenshot
Via Web UI:	User Part / Source: select 'Call E.164' Host Part / Source: select 'Local host name'

Actions	Screenshot
	 Via CLI: context cs SWITCH interface sip <sip_interface_to_OBS> address-translation outgoing-call identity-header user-part call e164 single-user host-part local
Repeat the same operation for all the SIP Interfaces towards OBS	.

5.4.4 Numbers Manipulations

This chapter is about the number manipulation for precisely the “Called Number” in the URI. OBS Phone numbers must be sent to Orange in E164 format.

The following example manipulations will transform Called Numbers received from Customer IPPBX in National format (0ZABPQMCDU or 00xxxxxxxx) to E164 (+CCZABPQMCDU) before sending the Call towards Orange BTALK.

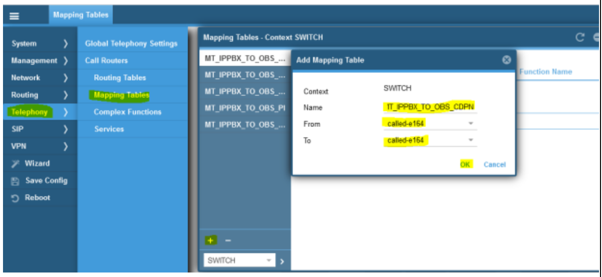
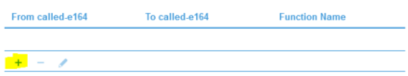
Note: *+CC prefix is the Country Code of the country where the SBC or IPPBX is installed. It is up to the Customer to indicate the correct +CC. ex +33 for France. If the IPBX is using a local dial plan (Private numbering Plan), then the manipulation has to adapted in consequence by the Customer.*

OBS BTalk Transformations

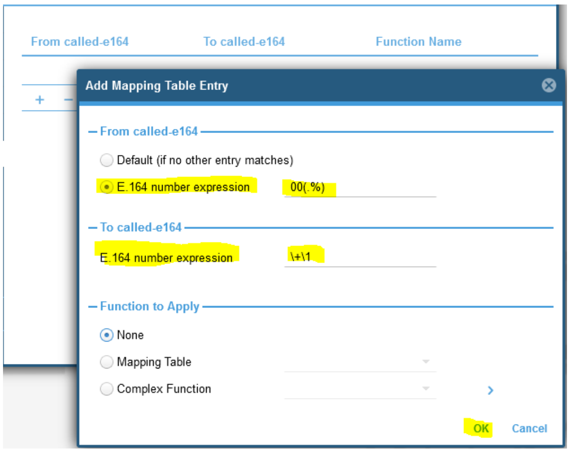
MT order	Mapping-Table name	Input Type of MT	Output Type of MT	Input Value	Output Value
1	MT_IPPBX_TO_OBS_CDPN	called-e164	called-e164	00(.%) 0(.%)	\+\1 \+33\1
2	MT_IPPBX_TO_OBS_CNTN	calling-type-of-number	calling-type-of-number	default	international
3	MT_IPPBX_TO_OBS_CNPN	calling-e164	calling-e164	00(.%) 0(.%) (....)	\1 33\1 33ZABPQ\1

1. mappings for called number normalization / transformation (00 → E164, 0 →E164) .
2. setting the calling type of number to international will add the leading '+' to the calling number. This is why we do not add the leadnig '+' in the mappings in step 3.
3. mapping table in case of internal format of calling number from IPPBX (without leading +CCZABPQ…. or 0ZABPQ….) or in case of 0ZABPQMCDU or 00CCZABPQMCDU calling number format. Important: in the example with 33ZABPQ\1 replace this REGEX by the IPPBX installation number, for example: 3329608\1

Called Party Number (00 > E164 and 0 > E164)

Actions	Screenshot
1. Create MT_IPPBX_TO_OBS_CDPN Mapping Table (transformation of called E164 number)	Via Web UI: Open the menu Telephony > Mapping Tables, then click on '+' in the bottom left corner to create a new Mapping Table. Name: enter 'MT_IPPBX_TO_OBS_CDPN' From: select 'called-e164' To: select 'called-e164' Confirm with OK  Via CLI: context cs mapping-table called-e164 to called-e164 MT_IPPBX_TO_OBS_CDPN
2. Create Mapping Table entries	Select the previously created Mapping Table MT_IPPBX_TO_OBS_CDPN, then click on '+' button on the right side of the window (table entries) to create a new mapping table entry:  In the new window configure the following: From called-E164/E164 number expression: enter 00(.%)

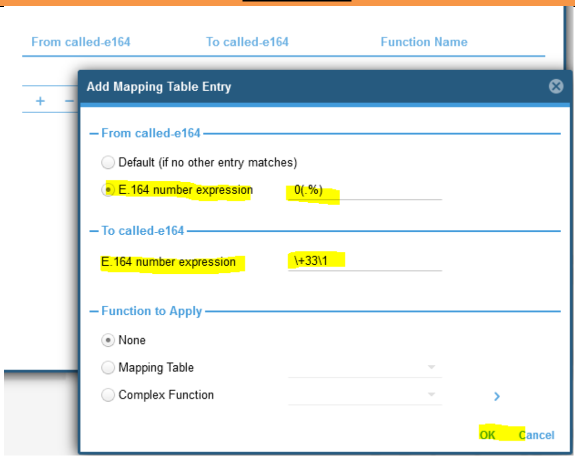
<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>
To called-E164/E164 number expression: enter \+\1
Confirm with OK



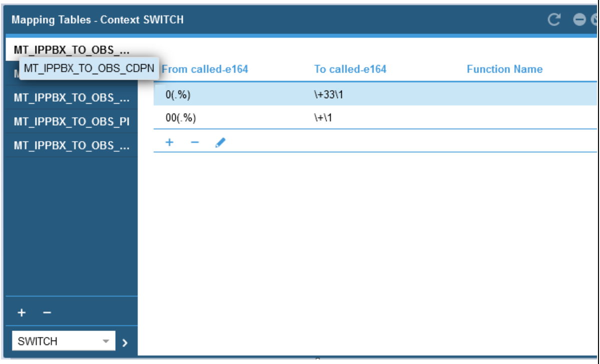
Via CLI:

```
context cs
mapping-table called-e164 to called-e164 MT_IPPBX_TO_OBS_CDPN
  map 00(.%) to \+\1
```

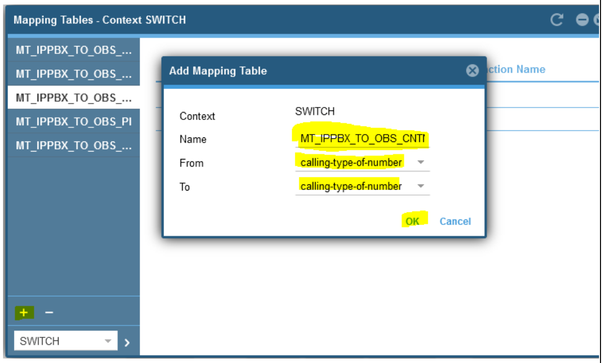
Similar operation for the 2nd mapping table entry: click on ‘+’ button on the right side of the window (table entries) to create a new mapping table entry.
From called-E164/E164 number expression: enter 0(.%)
To called-E164/E164 number expression: enter \+33\1
Confirm with OK
</td></tr>
</table>

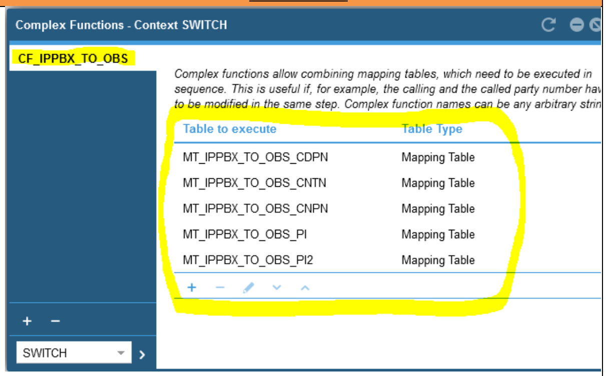
<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>

Via CLI:
<code>context cs</code>
<code>mapping-table called-e164 to called-e164 MT_IPPBX_TO_OBS_CDPN</code>
<code> map 0(.%) to \+33\1</code>

Mapping table entries after the creation of these two entries:

</td></tr>
<tr><td>3. Call the created Mapping Table from the Complex Function CF_IPPBX_TO_OBS</td><td>See how to proceed at the end of chapter Mapping Table</td></tr>
</table>

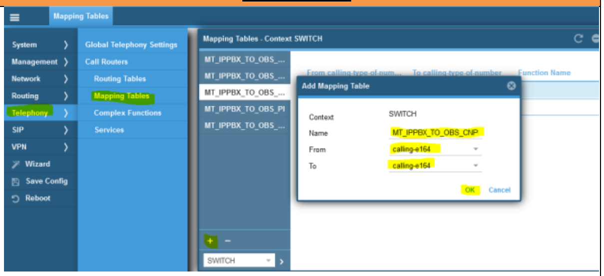
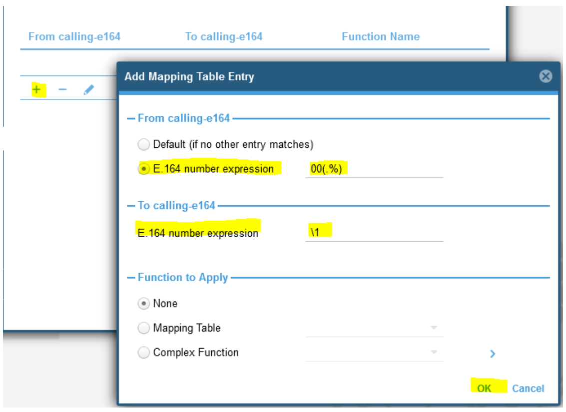
Calling Type of Number Transformation

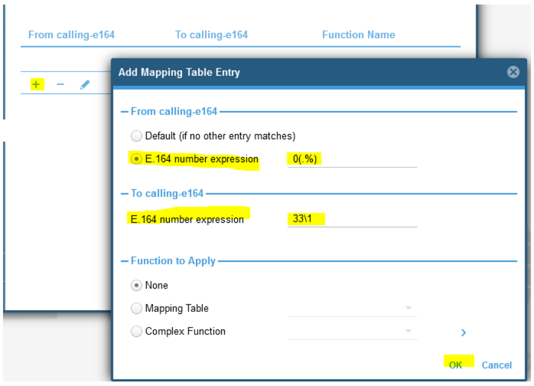
Actions	Screenshot
1. Create MT_IPPBX_TO_OBS_CNTNM apping Table (transformation of calling type of number)	Via Web UI: Open the menu Telephony > Mapping Tables, then click on '+' in the bottom left corner to create a new Mapping Table. Name: enter 'MT_IPPBX_TO_OBS_CDTN' From: select 'calling-type-of-number' To: select 'calling-type-of-number' Confirm with OK  Via CLI: context cs mapping-table calling-type-of-number to calling-type-of-number MT_IPPBX_TO_OBS_CNTN
2. Create Mapping Table entries	Only via CLI (bug identified in Web UI for this type of Mapping Table) `context cs` `mapping-table calling-type-of-number to calling-type-of-number MT_IPPBX_TO_OBS_CNTN` `  map default to international`
3. Call the created Mapping Table from the Complex Function CF_IPPBX_TO_OBS	See how to proceed at the end of chapter Mapping Table When all mentioned Mapping Tables have been added to CF_IPPBX_TO_OBS, the complex function should have the following content: Web UI:

Actions	Screenshot
	 The displayed order of the MT's is not mandatory in this use case. CLI: `context cs SWITCH` `  complex-function CF_IPPBX_TO_OBS` `    execute 1 MT_IPPBX_TO_OBS_CDPN` `    execute 2 MT_IPPBX_TO_OBS_CNTN` `    execute 3 MT_IPPBX_TO_OBS_CNPN` `    execute 4 MT_IPPBX_TO_OBS_PI` `    execute 5 MT_IPPBX_TO_OBS_PI2` The displayed order of the MT's is not mandatory in this use case.

Calling Party Number Transformation

Actions	**Screenshot**
1. Create MT_IPPBX_TO_OBS_CNPNMapping Table (transformation of calling E164 number)	Via Web UI: Open the menu Telephony > Mapping Tables, then click on '+' in the bottom left corner to create a new Mapping Table. Name: enter 'MT_IPPBX_TO_OBS_CNPN' From: select 'calling-e164' To: select 'calling-e164' Confirm with OK

Actions	Screenshot
	 Via CLI: `context cs` `mapping-table calling-e164 to calling-e164` `MT_IPPBX_TO_OBS_CNPN`
2. Create Mapping Table entries	Select the previously created Mapping Table MT_IPPBX_TO_OBS_CNPN, then click on '+' button on the right side of the window (table entries) to create a new mapping table entry:  In the new window configure the following: From calling-E164/E164 number expression: enter 00(.%) To called-E164/E164 number expression: enter \1 Confirm with OK  Via CLI: `context cs` `mapping-table calling-e164 to calling-e164`

Actions	Screenshot
	`MT_IPPBX_TO_OBS_CNPN` `  map 00(.%) to \1` Similar operation for the 2nd mapping table entry: click on '+' button on the right side of the window (table entries) to create a new mapping table entry. In the new window configure the following: From calling-E164/E164 number expression: enter 0(.%) To called-E164/E164 number expression: enter 33\1 Confirm with OK  Similar operation for the 3rd mapping table entry: click on '+' button on the right side of the window (table entries) to create a new mapping table entry. In the new window configure the following: From calling-E164/E164 number expression: enter (....) To called-E164/E164 number expression: enter CCZABPQ\1 Replace CCZABPQ\1 by the IPPBX installation number, for example: 3329608\1 Confirm with OK
3. Call the created Mapping Table from the Complex Function CF_IPPBX_TO_OBS	See how to proceed at the end of chapter Mapping Table

5.4.5 Outbound Manipulations

At the egress, SIP messages already processed by the SBC are modified to meet the SIP requirements of the upstream device.

Diversion header – outgoing calls to OBS

Sending out of Diversion header is not enabled by default.

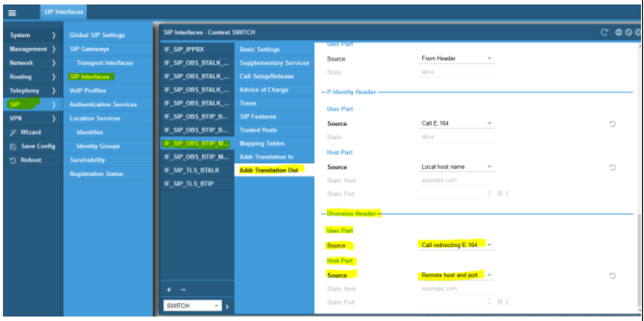
For enabling sending of the Diversion Header, an outgoing address translation expression must be configured on the SIP Interface. This expression specifies how to create the Diversion URI of the header.

As User Part of the URI, the Calling Redirecting number of the internal Call Router will always be taken. The user must configure the Host Part that is set per default to none. Setting the Host Part to none disables transmission of the Diversion Header.

The following header manipulation is necessary:

SIP Interface	Parameter	Value
<sip_interface_to_OBS>	address-translation outgoing-call diversion-header user-part	redir host-part remote

On all SIP Interfaces configured towards BTIP, BTalk, BTIPoI, BToI proceed to the following configuration: go to the Web UI menu SIP > SIP Interfaces, choose one of the SIP Interfaces configured towards BTIP, BTalk, BTIPoI, BToI, then select the submenu 'Addr. translation Out' and under the configuration part 'Diversion header' select the following settings:

Actions	**Screenshot**
Via Web UI	User Part / Source: select 'Call redirecting E.164' Host Part / Source: select 'Remote host and port'  Via CLI: **context cs SWITCH** **interface sip <if_sip_name>** **address-translation outgoing-call diversion-header user-part redir host-part remote**
Repeat the same operation for all the SIP Interfaces towards OBS	

5.4.6 Inbound Manipulations

At the ingress, inbound SIP messages are modified to permit proper handling by the SBC's routing function.

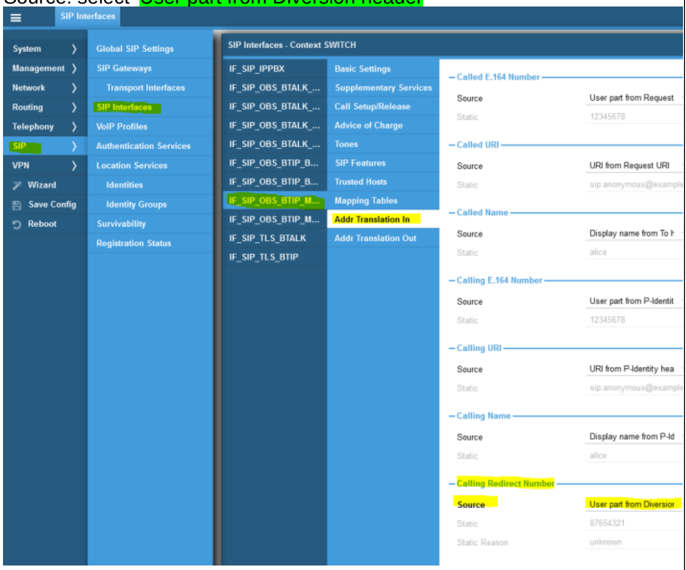
Diversion header – incoming calls from OBS

For receiving of the Diversion Header, an incoming address translation expression must be configured on the SIP Interface. Because several methods for transmitting redirecting information are available, this expression specifies that they must be taken from the Diversion Header when providing them to the internal call control.
The following header manipulation is necessary:

SIP Interface	Parameter	Value
<sip_interface_to_OBS>	address-translation incoming-call calling-redir	diversion-header

On all SIP Interfaces configured towards BTIP, BTalk, BTIPoI, BToI proceed to the following configuration: go to the Web UI menu SIP > SIP Interfaces, choose one of the SIP Interfaces configured towards BTIP, BTalk, BTIPoI, BToI, then select the submenu 'Addr. translation In' and under the configuration part 'Calling Redirect Number' select the following settings:

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td>Via Web UI</td><td>Source: select 'User part from Diversion header'



Via CLI:

```
context cs SWITCH
interface sip <sip_interface_to_OBS>
  address-translation incoming-call calling-redir diversion-header
```
</td></tr>
</table>

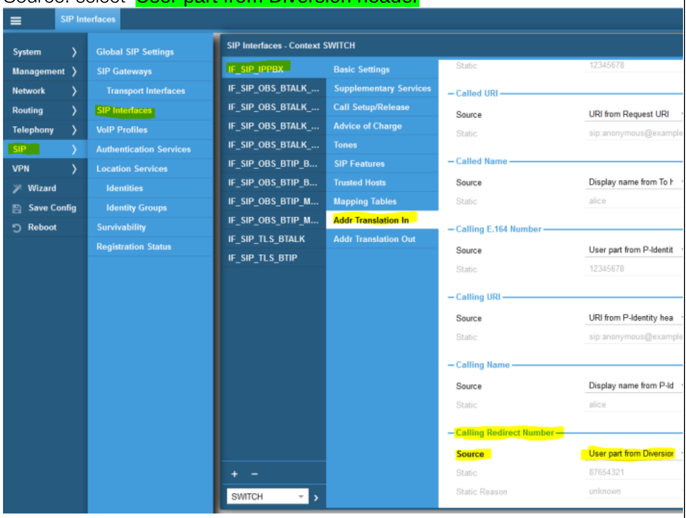
Actions	Screenshot
Repeat the same operation for all the SIP Interfaces towards OBS.	

Diversion header – incoming calls from IPPBX

In order to allow proper handling of the received Diversion Header from IPPBX by the SBC's routing function, an incoming address translation expression must be configured on the IPPBX Interface.

SIP Interface	Parameter	Value
<sip_interface_to_IPPBX>	address-translation incoming-call calling-redir	diversion-header

On the SIP Interface configured towards the IPPBX proceed to the following configuration: go to the Web UI menu SIP > SIP Interfaces, choose the SIP Interface IF_SIP_IPPBX, then select the submenu 'Addr. translation In' and under the configuration part 'Calling Redirect Number' select the following settings:

Actions	Screenshot
Via Web UI	Source: select 'User part from Diversion header'  Via CLI: `context cs SWITCH` `interface sip <sip_interface_to_IPPBX>` `  address-translation incoming-call calling-redir diversion-header`
Repeat the same operation for all the SIP Interfaces towards OBS.	

Annexes

5.5 Example of SIP INVITE message

From IPPBX towards Orange BTALK

```
INVITE sip:+33399102573@172.22.244.209:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 6.6.77.10:5060;branch=z9hG4bK84b0c101e26cd83fa
Max-Forwards: 70
From: 33296082204 <sip:+33296031504@6.6.77.10;user=phone>;tag=0b27086b73
To: <sip:+33399102573@172.22.244.209:5060;user=phone>
Call-ID: 5542c29c99df9c02
CSeq: 688798394 INVITE
Allow: ACK, BYE, CANCEL, INVITE, NOTIFY, OPTIONS, UPDATE
Contact: <sip:+33296031504@6.6.77.10:5060;transport=udp>
P-Asserted-Identity: 33296082204 <sip:+33296031504@6.6.77.10;user=phone>
P-Early-Media: supported
Session-Expires: 1800
Supported: timer, replaces
User-Agent: XiVO PBX 2021.07.02, Patton SN500 00A0BA10DD86 3.20.2-21122
Content-Type: application/sdp
Content-Length: 250

v=0
o=MxSIP 0 6258 IN IP4 6.6.77.10
s=SIP Call
c=IN IP4 6.6.77.10
t=0 0
m=audio 6410 RTP/AVP 8 18 101
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:18 annexb=no
a=fmtp:101 0-16
a=ptime:20
a=sendrecv
```

From Orange BTALK toward Customer IPPBX

```
INVITE sip:+33296082204@6.6.77.10:5060;user=phone SIP/2.0
Via: SIP/2.0/UDP 172.22.244.209:5060;branch=z9hG4bKkm4mii008oaoe73idre0.1
Max-Forwards: 64
From: "+3341319852573" <sip:+3341319852573@172.22.244.209;user=phone>;tag=SDvl5md01-U0egvw
To: <sip:+33296082204@6.6.77.10;user=phone>
Call-ID: SDvl5md01-c6cf5be2591ef5f782a5ce03cefbc16f-v300g000I0
CSeq: 416962 INVITE
Contact: <sip:172.22.244.209:5060;transport=udp>
Supported: em,path,resource-priority,sdp-anat
Allow: INVITE, ACK, CANCEL, BYE, UPDATE, INFO, OPTIONS, REFER
Privacy: none
Content-Type: application/sdp
Content-Length: 265
P-Charging-Vector: icid-value=e5a5dea5-3a43-4def-8ff0-da1460665d5b

v=0
o=- 1694515214 1510818940 IN IP4 172.22.244.209
s=-
c=IN IP4 172.22.244.209
t=0 0
m=audio 14696 RTP/AVP 8 18 101
a=fmtp:18 annexb=no
a=rtpmap:8 PCMA/8000
a=rtpmap:18 G729/8000
a=rtpmap:101 telephone-event/8000
a=fmtp:101 0-15
a=sendrecv
a=ptime:20
```

5.6 Set a superuser account

All Patton eSBCs are delivered from factory with a default user 'admin' and an empty password. Putting into service the unit with this default admin account could seriously compromise the security of the unit.

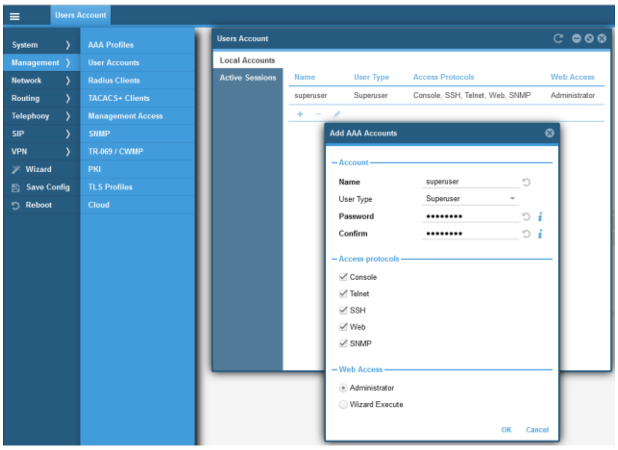
There are three user types on Patton eSBCs with different levels of privileges. These are:

- superuser: with full access
- administrator: with full access (no rights to create new users)
- operator: with restricted access

Therefore, it is strongly recommended to set a superuser account straight after the initial bootup and keep in mind or save the credentials. After the creation of a first superuser account, the initial admin account (which was also a superuser account type, with username 'admin') is automatically removed and replaced by the newly created superuser.

<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td>Set a superuser account</td><td>Note: in the given example we set a new user of type „Superuser“ (highest user privileges), with the name superuser. You can set a different user name, but it is important to set this user type.

Web UI:



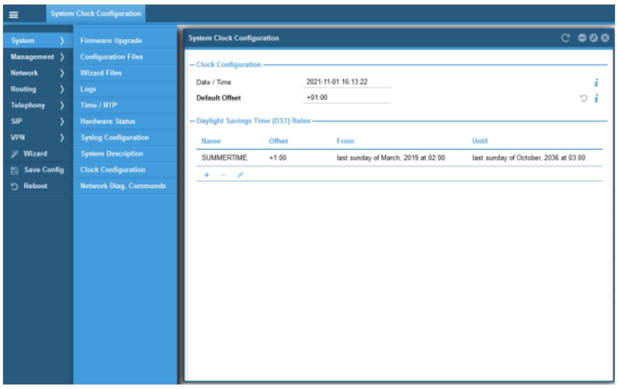
Via CLI:

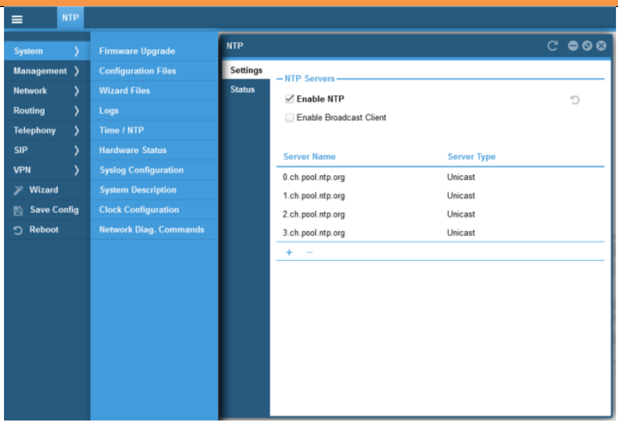
<code>superuser <user_name> password <your_password></code></td></tr>
</table>

5.7 NTP server configuration

This section describes how to configure the NTP server's IP address. It is recommended to implement an NTP server (Microsoft NTP server or another global server) to ensure that the SBC receives the current date and time. This is necessary for validating certificates of remote parties. It is important, that the NTP Server is located on the LAN IP Interface and accessible through it.

To configure the NTP server address:

Actions	Screenshot
1. Set time zone	The following example correponds to CET time zone (Central European Time) with DST. Web UI:  Via CLI: `clock local default-offset +01:00` `clock local dst-rule SUMMERTIME +1:00 from mar last sunday 02:00 2019 until oct last sunday 03:00 2036`
2. NTP Server Configuration	Web UI: configure the preferred NTP server pool according to your environment. The provided example corresponds to a public NTP pool for the specific pool zone of Switzerland. In your customer environment you might have to use a local NTP server / pool, i.e. located in the LAN network and accessible through the LAN interface of the SBC.

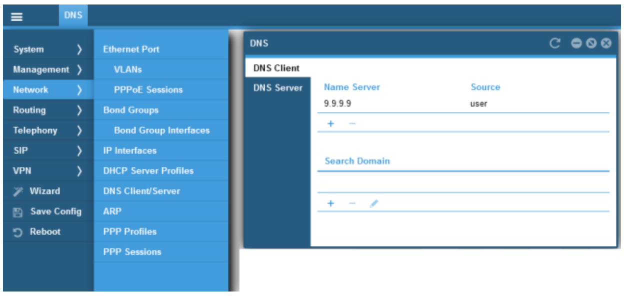
<table>
<tr><th>Actions</th><th>Screenshot</th></tr>
<tr><td></td><td>


Via CLI:

```
ntp
  server 0.ch.pool.ntp.org
  server 1.ch.pool.ntp.org
  server 2.ch.pool.ntp.org
  server 3.ch.pool.ntp.org
  no shutdown
```
</td></tr>
</table>

To get further information about configuring NTP time source, please consult Patton Trinity CLI Reference Guide (see References documents) -> Chapter “NTP Client Configuration”.

5.8 DNS Server configuration

Actions	Screenshot
DNS Server DNS Server configuration seen from Patton SBC means you have to configure Patton's DNS Client side.	Web UI: In the lab, as shown in the example below, we used the public DNS Server with IPv4 address 9.9.9.9. You should configure private/public DNS server(s) according to your IT environment.  Via CLI: `dns-client` `  name-server 9.9.9.9`

5.9 eSBC local security ACL

As already mentioned earlier at the end of the Chapter Patton Global Configuration / Configure Network Interfaces, in the topology for BToI / BTIPoI the WAN IP Interface of Patton SBC is interconnected through the enterprise DMZ behind a firewall. Additionally, on Patton eSBC level, an Access Control List can be applied, which allows only BToI/BTIPoI-relevant traffic in order to avoid attacks from the internet.

Actions	Screenshot
Create ACL profile	Only via CLI: `profile acl ACL_WAN_TLS` `  permit 1 src-ip <BT_Nominal_IP>` `  permit 2 src-ip <BT_Backup_IP>` `  deny 2`
Apply ACL profile to WAN_TLS network interface in incoming direction	Only via CLI: `context ip` ` interface WAN_TLS` `   use profile acl in ACL_WAN_TLS`

Note: the provided ACL example allows incoming IP traffic only from the defined IP addresses, without any protocol restriction. Additionally, it is possible to restrict incoming traffic only to a certain TCP destination port, typically to TCP/5061 for this scenario. In this case the required CLI command is the following:

```
permit 1 protocol tcp src-ip <BT_Nominal_IP> dest-port 5061
permit 1 protocol tcp src-ip <BT_Backup_IP> dest-port 5061
```

Commenté [CSO6]: Remplacer les @ IP par <BT_Nominal_IP> & <BT_BackupI_IP>

Commenté [CSO7]: Remplacer les @ IP par <BT_Nominal_IP> & <BT_BackupI_IP>

Glossary

A : DNS Address record

BTalk: Business Talk

BTIP: Business Talk IP

CC: Country Code

CSBC/ESBC: Customer/Enterprise Session Border Controller

CSR: Certificate Signing Request

DTMF: Dual Tone Multi Frequency

FQDN: Fully Qualified Domain Name

IP: Internet Protocol

LAN: Local Area Network

LLDP: Link Layer Discovery Protocol

MMS: Message Manipulation SIP

NET: Network Equipment Technologies

PBX: Private Branch eXchange

PSTN: Public Switched Telephone Network

RS: Remote Site

SBC: Session Border Controller

SDP : Session Description protocol

Sg : Signaling group

SIP: Session Initiation Protocol

SRTP: Secure Real Time Protocol

SRV : DNS Service record

TCP: Transmission Control Protocol

TLS: Transport Layer Security

UDP: User Datagram Protocol

WAN: Wide Area Network