

Frafos ABC SBC Microsoft Direct Routing configuration guide

Frafos GmbH

Contents

Preface	3
Planning the connection with Microsoft Direct Routing	4
Prepare FQDN for the SBC	4
Prepare a certificate for the SBC	4
Initial SBC's configuration	6
License	6
G729 codec	6
Firewall	6
Global configuration options	6
TLS profiles	7
Call agents	8
Primary call agent	8
Backup call agents	10
SIP/trunk call agents	11
Routing rules	12
A/C rules	14
Set SBC's FQDN Contact and From header	14
Reply to OPTIONS messages	15
Handling INVITE with Replaces	16
Update Supported header	16
Call transfers	16
RTP anchoring and RTCP generation	17
Force SRTP towards Microsoft Direct Routing	18
Add R-URI parameter user=phone	18
Blacklist extmap SDP attribute	19
Add rtcp-mux SDP attribute	19
DTMF mediation	19
Call agent specific rules	21
Optional SIP Mediation	22
Transcoding	23
Transcoding on incoming call leg	23
Transcoding on outbound call leg	24
Troubleshooting	27
Monitoring pages	27
Frafos ABC monitor	27
Traffic logging	27

SEMS log 27

Decrypt TLS traffic with wireshark 28

References **29**

Preface

This document describes the main configuration aspects of ABC SBC when used together with Microsoft Direct Routing. It contains only Microsoft Direct Routing related configuration changes and it expects the system was already installed and the basic configuration has been done as well. Please refer to the ABC SBC handbook to cover installation and initial configuration steps.

Planning the connection with Microsoft Direct Routing

Prepare FQDN for the SBC

Microsoft Direct Routing proxies do not accept IP but require a FQDN in SBC's host parts of Contact and From URIs. This FQDN needs to correspond to domain configured in [Microsoft 365 admin center](#), see [here](#).

Prepare a certificate for the SBC

According to [this](#) it is necessary to prepare a public trusted certificate for the SBC.

There can be more ways how to obtain a certificate, depeneding on the certification authority. It is quite common, that the certification authority processes a certificate signing request (CSR) created by the client and generates the certificate with appropriate content.

The CSR must contain common name (CN) that should be filled with SBC's FQDN.

Additionally, it is recommended to add Subject Alternative Name (SAN) where the FQDN and even IP address(es) of the SBC are listed.

Optionally, the CSR can contain other fields, the most common are:

- country (C)
- state (ST)
- locality name (L)
- organization name (O)
- organization unit name (OU)

To generate the CSR it is possible to use `openssl` like in following example.

```
openssl req -new -out sbc.csr -newkey rsa:2048 -nodes -sha256 \
    -keyout sbc.key -config csr.conf
openssl rsa -in sbc.key -out sbc_key.pem
openssl req -text -noout -verify -in sbc.csr -key sbc_key.pem
```

The CSR properties are specified in [csr.conf](#) config file, that can look like:

```
[req]
distinguished_name = req_distinguished_name
req_extensions = v3_req
prompt = no

[req_distinguished_name]
C = DE
ST = Germany
```

```
L = Berlin
O = Frafos
CN = teams.frafos.net
```

```
[v3_req]
subjectAltName = @alt_names
```

```
[alt_names]
DNS.1 = teams.frafos.net
IP.1  = 192.168.13.220
```

The generated CSR content can be verified again by openssl:

```
openssl req -in sbc.csr -text -noout
```

Initial SBC's configuration

License

Following items need to be enabled in a license to allow particular SBC features that are necessary or recommended when connecting Frafos ABC SBC with Microsoft Direct Routing.

- media server

Media server license is needed for “Activate Inband DTMF Detection” action used for conversion of in-band DTMF to AVT. This is necessary if SIP devices used in the network do not offer “telephone-event” payload according to RFC 2833.

- transcoding

Transcoding license is necessary if there are SIP devices in the network that do not support codecs offered by Microsoft Direct Routing proxies (see [Media traffic: Codecs](#)) and ABC SBC is expected to transcode media for them using the “Activate transcoding” action.

G729 codec

For g729 codec support `sems-g729` package needs to be installed manually after installing the SBC.

```
yum install sems-g729
```

Firewall

According to [Plan direct routing article](#) the possible source subnets for traffic coming from Microsoft Direct Routing proxies are 52.112.0.0/14 and 52.120.0.0/14.

If there are manual firewall rules configured, make sure that these subnets are allowed.

Global configuration options

There are several options that need to be set in Config -> Global Config menu.

For interoperability with Microsoft Direct Routing, the option “Enable Microsoft Direct Routing” on “SEMS” tab must be enabled.

The “Minimal supported TLS version, use tls1 / tls1.1 / tls1.2.” on “SEMS” tab must be set to “tls1.2” to avoid usage of deprecated TLS versions.

To ease troubleshooting the SBC should be connected to Frafos ABC Monitor. In such case its IP address needs to be given on “Events” tab.

Additionally, for easier troubleshooting, following options should be set on “SEMS” tab:

- “Dump TLS session keys to file” should be checked. This causes that session keys used for signaling traffic will be written into a file (`/data/pcap/tls_keys/signaling`) and can be used for example by Wireshark to decrypt network traffic captures made by hand.
- “Log level” should be set to 3. This means that SEMS (the process handling signaling and media traffic) will be logging at debug level. This settings is not intended for production environment but is highly recommended for testing.

TLS profiles

TLS profile used on signaling interface for communication with Microsoft Direct Routing proxies needs to use certificate with SBC's FQDN in CN or Subject Alternative Name (see [here](#)).

The “Trusted CA certificates file” in the TLS profile must contain bundle of SBC certificate issuer's chain and the CA certificates used for signing certificates of Microsoft Direct Routing proxies. For this purpose:

- download [certificate bundles used by Microsoft Direct Routing proxies](#)
- convert them to PEM

```
openssl pkcs7 -print_certs -in m365_root_certs_20201012.p7b -inform der \
    -out m365_root_certs_20201012.pem
openssl pkcs7 -print_certs -in m365_intermediate_certs_20201013.p7b -inform pem \
    -out m365_intermediate_certs_20201013.pem
```

- put it all together with another required CA certificates (SSL.com ones here because the SBC's certificate is from SSL.com) into one bundle of trusted certificates

```
cat m365_root_certs_20201012.pem \
    m365_intermediate_certs_20201013.pem \
    SSL_COM_ROOT_CERTIFICATION_AUTHORITY_RSA.crt \
    SSL_COM_RSA_SSL_SUBCA.crt > trusted_bundle.pem
```

If there are another TLS peers in the network, the bundle with trusted CA certificates should be extended to contain CA certificates (or chains of them) used for signing the peer TLS certificates. Like

```
cat another_root_ca.crt another_signing.crt >> trusted_bundle.pem
```

The `trusted_bundle.pem` then should be used as “Trusted CA certificates file” in the appropriate TLS profile.

If all the peer CA certificates are present in the “Trusted CA certificates file”, then “Verify peer certificate” can be enabled in the TLS profile, to force peer's certificate verification.

Note that using “Verify peer certificate” option in “default” TLS profile is **not recommended**. Default profile may be used as backup if there is no other TLS profile configured (for example for configuration pull) and doing a mistake in configuration of the certificate on both sides may cause rejecting connection between them if “Verify peer certificate” is set.

If you plan to use this option it is highly recommended to specify TLS profiles one by one, explicitly, for specific configuration areas. For example one TLS profile for external management interface (GUI), another TLS profile for SIP signaling and yet another for WebRTC interface.

Call agents

On the SBC side it is necessary to configure call agents logically corresponding to Microsoft Direct Routing gateways. According to [this article](#) the connection points are following three FQDNs, ordered according to priority:

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

ms		insert call agent	A-rules	C-rules	
sip.pstnhub.microsoft.com Assigned to: Group:default	signaling	sip.pstnhub.microsoft.com	A-rules	C-rules	
sip2.pstnhub.microsoft.com Assigned to: Group:default	signaling	sip2.pstnhub.microsoft.com	A-rules	C-rules	
sip3.pstnhub.microsoft.com Assigned to: Group:default	signaling	sip3.pstnhub.microsoft.com	A-rules	C-rules	

Figure 1: Call agents

For proper configuration of the call agents it is necessary to consider that:

- sip.pstnhub.microsoft.com is the primary destination and all the signaling traffic needs to be sent there if the destination is available
- the other destinations should be used just for failover purposes and not for load balancing
- all the outbound destinations (call agents) need to be monitored by OPTIONS requests
- all destinations need to use TLS for signaling

In order to make the following documentation more readable we expect that there is a realm called “ms” and this realm contains all call agents necessary for communication with Microsoft Direct Routing proxies mentioned above.

Primary call agent

This call agent represents the top priority destination given by FQDN sip.pstnhub.microsoft.com.

For failover purposes the lower priority call agents need to be configured as “Backup call agent” resp. “2nd backup call agent”.

Destination monitoring needs to be properly configured, the Contact header has to use proper SBC’s FQDN. Additionally Supported header should be present, advertising support for “replaces” extension.

Name:	sip.pstnhub.microsoft.com		
Signaling interface:	signaling		
Media interface:	media		
Backup call agent:	sip2.pstnhub.microsoft.com		
2nd backup call agent:	sip3.pstnhub.microsoft.com		
Identified by:	Domain or Host Name		
Force transport:	<input checked="" type="checkbox"/>		
Transport protocol:	TLS		
Domain or Host Name	sip.pstnhub.microsoft.com	Port	
		Priority	Weight
		10	10
[Add destination]			
Assigned nodes and config groups:	Group: default x		

Destination Monitor	Blacklist Call Agent	Register Agent	Topology Hiding	Firewall Blacklisting	Interoperability	QoS
Monitoring interval:	60					
Max-Forwards:	0					
From URI:	sip:teams.frafos.net					
Destination URI:						
Additional header fields:	Contact: <sip:teams.frafos.net:5061;transport=tls>\r\nSupported: replaces					

Figure 2: Primary call agent

Blacklisting of the call agent upon monitoring failures or specific responses (503 by default) should be configured:

Destination Monitor	Blacklist Call Agent	Register Agent	Topology Hiding	Firewall Blacklisting	Interoperability	QoS
Blacklist TTL:	120					
Blacklist grace timer (ms):	32000					
Blacklist error reply codes:						
Destination Blacklist for in-dialog requests:	<input type="checkbox"/>					

Figure 3: Primary call agent: blacklisting

“MS-Teams compatibility mode” needs to be used on call agent’s Interoperability tab:

Destination Monitor
Blacklist Call Agent
Register Agent
Topology Hiding
Firewall Blacklisting
Interoperability
QoS

Contact-less REGISTER replies: ☐

Do not update next-hop: ☐

Authentication user for call transfer related requests:

Password for call transfer related requests:

Mediasec support: ☐

Hold settings:

Avoid 0.0.0.0 in connection address in SDP: ☐

MS-Teams compatibility mode: ☒

Allow UPDATE with SDP during O/A exchange: ☐

Figure 4: Primary call agent: interoperability

Backup call agents

Both the call agents for sip2.pstnhub.microsoft.com and sip3.pstnhub.microsoft.com are configured the same way, just the FQDN differs. In this case no backup call agent is configured.

Name:	sip2.pstnhub.microsoft.com		
Signaling interface:	signaling		
Media interface:	media		
Backup call agent:			
2nd backup call agent:			
Identified by:	Domain or Host Name		
Force transport:	<input checked="" type="checkbox"/>		
Transport protocol:	TLS		
Domain or Host Name	sip2.pstnhub.microsoft.com	Port	
		Priority	Weight
		10	10
[Add destination]			
Assigned nodes and config groups:	Group: default x		

Destination Monitor
Blacklist Call Agent
Register Agent
Topology Hiding
Firewall Blacklisting
Interoperability
QoS

Monitoring interval:	60
Max-Forwards:	0
From URI:	sip:teams.frafos.net
Destination URI:	
Additional header fields:	Contact: <sip:teams.frafos.net:5061;transport=tls>\r\nSupported: replaces

Figure 5: Backup call agent

The **destination monitoring**, **blacklisting** and **interoperability** parameters of these call agents need to be configured the same way as in the sip.pstnhub.microsoft.com call agent.

SIP/trunk call agents

Call agents for other stuff depend on specific needs and must be handled case by case. For typical use cases please refer to the ABC SBC handbook.

Routing rules

There can be many ways how to configure routing:

- based on user part of the called URIs
- based on domain part of the called URIs
- based on prefixes of called numbers (for example E.164 URIs vs. local numbers)
- based on source (calls initiated by Microsoft Direct Routing proxies can be routed towards SIP/PSTN users, calls from SIP/PSTN side may be routed to Microsoft Direct Routing proxies)

It depends on the actual needs and must be handled case by case. For typical use cases please refer to the ABC SBC handbook.

In following example SBC routes numbers that are present in the table `teams_users` towards Microsoft Direct Routing gateways. Note that the primary call agent is used here.

Conditions

Match on: **Operator:** **Value:**

Route to

Route using:

Realm:

Call Agent:

Routing method

☒ Set next hop

☐ Route via R-URI

☐ Use another destination instead of CAs' destination(s)

☐ Use on first request only

☐ Update R-URI host

☐ Add Route header field

Advanced

☐ Replace DNS name in R-URI through the resolved IP address

☐ Update To-URI host

☐ Replace To-URI host with the resolved destination IP address

☐ Force transport

☐ Enable redirect handling

Rule is active: ☒

Figure 6: Route to Microsoft Direct Routing proxies

There is [one requirement](#) related to call transfers:

All transfers that use an SIP Refer message must go through the Microsoft Teams infrastructure. When the Microsoft SIP proxy sends an SIP Refer message to SBC, an SIP Invite message should be returned to the SIP proxy, not to PSTN or to any other destination. It is true even if the call is transferred to an external PSTN number.

This can be achieved by routing rule similar to the one shown in picture below, that is matched *before* other routing rules. Please note, that additional conditions or another routing rule may be necessary to properly handle even call transfers initiated by SIP side:

Conditions

Match on:

Operator:

Value:

Request source

is

call transfer

Add condition

Route to

Route using

Static route

Realm

ms

Call Agent

sip.pstnhub.microsoft.com

Routing method

☒ Set next hop

☐ Use another destination instead of CAs' destination(s)
 ☐ Use on first request only
 ☐ Update R-URI host
 ☐ Add Route header field

☐ Route via R-URI

Advanced

☐ Replace DNS name in R-URI through the resolved IP address
 ☐ Update To-URI host
 ☐ Replace To-URI host with the resolved destination IP address
 ☐ Force transport
 ☐ Enable redirect handling

Rule is active:

☒

Figure 7: Route transferred calls

A/C rules

The call agents used for Microsoft Direct Routing share most of the A/C rules, thus it is beneficial to specify these rules just once at realm level.

Set SBC's FQDN Contact and From header

Microsoft Direct Routing proxies require host part of Contact and From headers being SBC's FQDN. Create a rule in A and C rules of the ms realm containing "Set Contact-URI host" and "Set From host" actions like following.

Action:	Value:	Description:
Set Contact-URI host	<input type="text" value="teams.frafos.net"/>	↓ ✕ Set the Contact-HF URI host part used for the dialog
Set From host	<input type="text" value="teams.frafos.net"/>	↑ ✕ Set (override) the From host or hostport part

Figure 8: SBC's FQDN in Contact and From headers

There is another possibility how to set the FQDN in Contact header towards Microsoft Direct Routing proxies: it can be configured as "Public IP address" of appropriate signaling interface. This method should be used if there are multiple signaling interfaces configured on the SBC and one of them is dedicated for communication with Microsoft Direct Routing.

Owner type:	Node
Owner:	teams
System interface:	eth0
Type of IP address:	Autoconfig
Type of public IP address:	Manual
Public IP address:	teams.frafos.net
TLS profile:	teams

Figure 9: SBC's FQDN via Public IP address

Reply to OPTIONS messages

Monitoring OPTIONS requests sent by Microsoft Direct Routing proxies towards the SBC need to be replied with 200 OK. Contact header in the generated reply needs to be SBC's FQDN again. It is necessary to add "Supported" header and "Allow" header into this response to advertise support for the "replaces" extension and SIP REFER method.

To simplify Allow header manipulation, that is done from several places in the configuration, it is beneficial to use cluster config parameters referenced by "%parameter-name%" in input fields like in following example.

Create a rule with "Reply to request with reason and code" action listing all the required headers ("Contact: <sip:teams.frafos.net:5061;transport=tls>\r\nSupported: replaces\r\nAllow: %ms_allow%") in A rules of the ms realm.

Conditions		Operator:		Value:		Description:	
Match on:	Method	==	OPTIONS	SIP Method			
<input type="button" value="Add condition"/>							
Actions		Value:		Description:			
Action:	Reply to request with reason and code					Refuse the call or request with a specific reason and code	
Code	200						
Reason	OK						
Header fields	Contact: <sip:teams.frafos.net:5061;transport=tls>\r\nSupported: replaces\r\nAllow: %ms Parameter detected: ms_allow, description: Allow header towards MS-Teams, default: ACK, BYE, CANCEL, INVITE, NOTIFY, OPTIONS, REFER, REGISTER, (Edit)						
Blacklist by firewall if repeated	<input type="checkbox"/>						

Figure 10: Reply monitoring OPTIONS

The **ms_allow** cluster config parameter should be defined in Config -> Define cluster config parameters, having a default value listing all methods allowed from Microsoft Direct Routing proxies ("ACK, BYE,

CANCEL, INVITE, NOTIFY, OPTIONS, REFER, REGISTER”).

Ordering:	1
Label:	Allow header towards MS-Teams
Parameter name:	ms_allow
Type:	string
Default Value:	ACK, BYE, CANCEL, INVITE, NOTIFY, OPTIONS, REFER, REGISTER

Figure 11: Parameter with Allow header value

Handling INVITE with Replaces

Microsoft Direct Routing proxies may use INVITE with Replaces header for implementing call transfers and/or call hold and unhold.

To handle this on SBC, a rule with “Handle INVITE with Replaces header” action needs to be created in A rules of ms realm.

Action:	Value:	Description:
Handle INVITE with Replaces header		✗ Handle Replaces header in an incoming INVITE: incoming call is connected to an existing call leg.

Figure 12: INVITE with Replaces handling

Update Supported header

Adding “replaces” into Supported header via “Update Supported header” action is necessary to advertise support for handling INVITE with Replaces to the Microsoft Direct Routing proxies. This needs to be done in both - A and C rules - of the ms realm.

Action:	Value:	Description:
Update Supported header	<div>Add tags</div> <div>replaces</div>	✗ Add, remove or set tags in Supported header. Supported since SBC 4.5.

Figure 13: Supported header

Call transfers

SBC is capable to handle in-dialog REFER message via “Call transfer handling” action with “Mode” set to “handle internally”. For interoperability reasons the “Update Allow header” action should be used to manipulate Allow appropriately so the peer can rely on REFER message being supported.

Action:	Value:	Description:
Call transfer handling		
Mode	Handle REFER internally	↓ × Configures handling of in-dialog REFER requests. These can be either passed through the SBC, handled locally within the SBC, or rejected.
Reconnect on all failures during unattended transfer	<input checked="" type="checkbox"/>	
Do not terminate after unattended transfer	<input checked="" type="checkbox"/>	
Update Allow header		
Action	Add tags	↑ × Add, remove or set tags in Allow header. Note that 'Add' will not add unless Allow header already exists, set via 'Set' or 'Default tags' are given.
Tags	REFER	
Direction	To A-Leg (caller leg)	
Apply on	On requests & replies	
Default tags if Allow header doesn't exist	%ms_allow%	
Parameter detected: ms_allow, description: Allow header towards MS-Teams, default: ACK, BYE, CANCEL, INVITE, NOTIFY, OPTIONS, REFER, REGISTER, (Edit)		

Figure 14: Call transfer handling

In case of attended transfer (REFER with Refer-To header including Replaces) the appropriate call legs are connected internally.

Unattended (blind) transfer (Refer-To header in REFER message doesn't contain Replaces) causes that SBC issues a new INVITE with content taken from Refer-To header.

The parameter “Reconnect on all failures during unattended transfer” allows SBC to reconnect the call back, if the unattended transfer fails; for example when the new INVITE is rejected.

According to [documentation](#) the option “Do not terminate after unattended transfer” should be used to avoid troubles that can be observed if SBC tries to terminate the transferring call leg before Microsoft Direct Routing proxy fully finishes the call transfer.

Additionally, based on the same document, all INVITEs generated as result of call transfer should go to Microsoft Direct Routing proxies that handle them appropriately even if the call is destined for the SIP side of the SBC.

This can be accomplished by a routing rule for requests where source is “call transfer” and using appropriate Microsoft Direct Routing call agent as destination.

It is important to handle call transfers in both - A and C - rules of the ms realm. The rule is the same except value of “Direction” parameter of “Update Allow header” action which is set to “To A-Leg (caller leg)” value for A rules and “To B-Leg (callee leg)” for C rules.

RTP anchoring and RTCP generation

SBC should be configured to stay in media path between SIP/PSTN users and Microsoft Direct Routing proxies. For this purpose “Enable RTP anchoring” action needs to be added in A and C rules of ms realm.

If necessary, RTCP can be generated by SBC if the “RTCP Generation” and “RTCP interval” are configured appropriately. Please note, that especially “RTCP interval” may need tuning based on SIP/PSTN UA behavior, the default value need not to be appropriate in all cases.

Additionally, due to compatibility problems, it is necessary to enable “Ignore ICE offer” option towards Microsoft Direct Routing proxies. In that case SBC will not try to negotiate media addresses using ICE protocol.

Action:	Value:	Description:
Enable RTP anchoring		✗ Forces media to visit the SBC. If symmetric option is turned on IP addresses in SDP are ignored and media are sent symmetrically back for safer NAT traversal. With 'intelligent relay' enabled, media can flow directly between UAs if they are behind the same NAT.
Force symmetric RTP for UAC	<input checked="" type="checkbox"/>	
Lock on addresses learned from RTP	<input type="checkbox"/>	
Enable intelligent relay	<input type="checkbox"/>	
Source-IP header field	<input type="text" value="X-ABC-Source-IP"/>	
Offer ICE-lite	<input type="checkbox"/>	
Ignore ICE offer	<input checked="" type="checkbox"/>	
Offer RTCP Feedback	<input type="checkbox"/>	
RTCP Generation	<input type="text" value="When no RTCP was received during the interval"/>	
RTCP Interval	<input type="text" value="0"/>	
Keepalive	<input type="text" value="global value"/>	
Keepalive method	<input type="text" value="global value"/>	
Timeout	<input type="text" value="global value"/>	

Figure 15: RTP anchoring

Force SRTP towards Microsoft Direct Routing

The Microsoft Direct Routing requires secure RTP with keys negotiated using SDP between SBC and its gateways. Create following rule in A and C rules of ms realm to achieve this goal.

Action:	Value:	Description:
Force RTP/SRTP		✗ Forces RTP or SRTP use in the selected call leg, or use SRTP with RTP fallback (Cisco method). This is direction dependent, i.e. different for A and C rules. Note that the Global Configuration parameter 'DTLS-SRTP' must be enabled in order to use 'DTLS'.
Force plain RTP	<input type="checkbox"/>	
Force secure RTP	<input checked="" type="checkbox"/>	
SRTP with RTP fallback	<input type="checkbox"/>	
Key Exchange Mechanisms	<input type="text" value="SDP"/>	

Figure 16: Force SRTP (SDP)

Add R-URI parameter user=phone

According to documentation the user=phone parameter in R-URI may speed up the call establishment with Microsoft Direct Routing proxies. To add this parameter add a rule with “Set RURI parameter” action into C rules of the ms realm.

Action:	Value:	Description:
Set RURI parameter	user	✖ Set request URI parameter
	phone	

Figure 17: Add user=phone R-URI parameter

Blacklist extmap SDP attribute

It is necessary to remove extmap attributes towards Microsoft Direct Routing because SBC does not control direction inside them. In case the value is different than in stream direction attribute in the SDP, Microsoft Direct Routing will respond with 488 and reason: Extension attribute direction is incompatible with the stream direction.

Create a following rule in A and C rules of the ms realm to remove the attribute.

Action:	Value:	Description:
Set SDP attribute blacklist	extmap	✖ Comma-separated list of SDP attributes to blacklist.

Figure 18: Blacklist extmap SDP attribute

Add rtcp-mux SDP attribute

If RTCP multiplexing is required, it is necessary to ensure, that the rtcp-mux media attribute is present in SDP towards Microsoft Direct Routing proxies. Create a following rule in A and C rules of ms realm for this purpose.

Action:	Value:	Description:
Insert or Replace SDP Media Attribute (on leg)		✖ Adds a new SDP media attribute or replaces its value if already exists. Acts on call leg, on all requests/replies.
Attribute name	rtcp-mux	
Media	audio	
Attribute value		
Replace with	<input type="checkbox"/>	

Figure 19: Add rtcp-mux SDP attribute

DTMF mediation

In typical environment it is not necessary to change DTMF handling on the SBC and leave endpoints to negotiate supported method by themselves.

On the other hand, SBC can be used to mediate DTMF transmission if necessary.

Microsoft Direct Routing gateways do not support DTMF method other than AVT (RFC 2833 / 4733). In case it is necessary, the conversion into AVT can be forced via “Relay DTMF as AVT RTP” action. To handle it in both directions (for inbound and outbound calls) it is necessary to have this action in A and in C rules as well.

If there are devices capable of sending in-band DTMF only, it is necessary to use additionally “Activate Inband DTMF Detection” action.

If the in-band DTMF detection is activated, SBC is able to translate the DTMF signals detected in audio stream into AVT or SIP INFO according to configuration but it is not able to filter the DTMF tones out of the audio stream and thus may cause the digits being received twice on the receiving side, if the receiver is capable to handle in-band and AVT/INFO DTMF as well.

Please note that in-band DTMF detection brings significant overhead to RTP processing and should be used with care.

In rare cases it might be necessary to limit the range of DTMF events. This can be achieved using action “Insert or Replace SDP Payload Attribute” as shown in the examples below.

Action:	Value:	Description:
Insert or Replace SDP Payload Attribute (on leg)		
Attribute name	<input type="text" value="fmtpt"/>	Adds a new SDP payload attribute (i.e. 'fmtpt') or replaces its value if already exists. Acts on call leg, on all requests/replies.
Media	<input type="text" value="audio"/>	
Codec	<input type="text" value="telephone-event"/>	
Attribute value	<input type="text" value="0-16"/>	
Replace with	<input checked="" type="checkbox"/>	
Relay DTMF as AVT RTP		
Direction	<input type="text" value="To A-Leg (caller leg)"/>	Relay DTMF as AVT RTP packets (RFC4733/RFC2833)
Activate Inband DTMF Detection		
Direction	<input type="text" value="To A-Leg (caller leg)"/>	Activate Inband DTMF Detection
Mode	<input type="text" value="If no RTP/AVT"/>	

Figure 20: DTMF mediation: A rule

Actions

Action:	Value:	Description:
Insert or Replace SDP Payload Attribute (on leg)		↓ × Adds a new SDP payload attribute (i.e. 'fmtp') or replaces its value if already exists. Acts on call leg, on all requests/replies.
Attribute name	fmtp	
Media	audio	
Codec	telephone-event	
Attribute value	0-16	
Replace with	<input checked="" type="checkbox"/>	
Relay DTMF as AVT RTP		↑ ↓ × Relay DTMF as AVT RTP packets (RFC4733/RFC2833)
Direction	To B-Leg (callee leg)	
Activate Inband DTMF Detection		↑ × Activate Inband DTMF Detection
Direction	To B-Leg (callee leg)	
Mode	If no RTP/AVT	

Figure 21: DTMF mediation: C rule

Call agent specific rules

It is recommended, to have a C rule for each of the outbound call agents, which sets “RURI host” and “To host” to corresponding destination’s FQDN.

Figure 22 displays three screenshots of the Frafos ABC SBC configuration interface, showing call agent specific rules for three different call agents: sip.pstnhub.microsoft.com, sip2.pstnhub.microsoft.com, and sip3.pstnhub.microsoft.com.

Each screenshot shows a table of rules for the selected call agent. The table has columns for Conditions, Actions, Continue, Active, and Comment. The rules are configured as follows:

Conditions	Actions	Continue	Active	Comment
<always>	Set RURI host: sip.pstnhub.microsoft.com, Set To host: sip.pstnhub.microsoft.com	✓	✓	

Figure 22: Call agent specific rules

Optional SIP Mediation

In some cases it might be necessary to fix SIP headers that cause troubles to Microsoft Direct Routing proxies. Frafos ABC SBC offers broad range of actions for this purpose. The simplest and often a sufficient one can be removal of problematic headers.

For example it may happen that User-Agent header content freely chosen by a SIP UA is not acceptable for Microsoft Direct Routing proxies. The header is not important for call handling and thus can be safely removed. It needs to be considered that header removal may be necessary not even for calls towards Microsoft Direct Routing but even for calls initiated by Microsoft Direct Routing. Otherwise, for example, BYE sent by the SIP UA terminating a call initiated by Microsoft Direct Routing may be rejected.

For this purpose we can create a rule with “Remove Header” action like following one in in A and C rules of the ms realm.

Action:	Value:	Description:
Remove Header	User-Agent	✗ Removes a header field if present in the original request. Enter the header name. This entry field is case-insensitive.

Figure 23: Header removal

For more use cases about SIP mediation please refer to the ABC SBC handbook.

Transcoding

It is highly recommended not to use transcoding unless necessary (there is significant overhead related to transcoding), but in case there are devices in SIP network that do not support [codecs supported](#) by Microsoft Direct Routing proxies, it is necessary to activate transcoding on the SBC.

Due to a SBC limitation, transcoding has to be always activated in A rules, i.e. in the rules handling incoming request. It will not work properly if the “Activate transcoding” action is used in C rules.

Transcoding on incoming call leg

Transcoding activation for incoming calls should be introduced in an A rule of the appropriate call agent or realm and can be conditioned for example by missing codec(s).

Following example activates transcoding if PCMU codec is not present in codecs offered by the device and allows transcoding between PCMU/8000, GSM/8000, G729/8000 and speex/8000 codecs (note that there is a GUI wizard that helps to choose the codecs from available ones).

Conditions			
Match on:	Operator:	Value:	Description:
Codecs	Do not contain	pcmu	✕ Codec value
Add condition			
Actions			
Action:	Value:	Description:	
Activate transcoding	PCMU/8000,GSM/8000,G729/8000,speex/8000	↓ ✕ Activate transcoding for list of codecs.	

Figure 24: Transcoding of incoming calls

Please note, that the SIP UA must use one of the codecs allowed for transcoding, otherwise transcoding is not possible. In other words, there has to be at least one common codec in use, known to the UA and to the SBC as well. For example if the SIP UA would use just opus codec, the rule above won't activate transcoding, because “opus” is not listed in the codecs for transcoding.

If SIP UA uses for example speex codec only, SBC will add the others to the offer towards Microsoft Direct Routing proxies so they can find compatible ones.

In the other direction SBC adds the codecs into answer to let the calling UA to choose the proper one.

Offer from SIP UA:

```
m=audio 4000 RTP/AVP 98 97 99 96
c=IN IP4 1.1.1.1
b=TIAS: 44000
a=rtcp: 4001 IN IP4 1.1.1.1
a=sendrecv
a=rtpmap: 98 speex/16000
a=rtpmap: 97 speex/8000
a=rtpmap: 99 speex/32000
a=rtpmap: 96 telephone-event/8000
a=fmtp: 96 0-16
```

Offer sent by SBC towards Microsoft Direct Routing:

```
m=audio 10432 RTP/SAVP 98 97 99 96 0 3 18
```



```
c=IN IP4 2.2.2.2
b=TIAS: 44000
a=rtpmap: 98 speex/16000
a=rtpmap: 97 speex/8000
a=rtpmap: 99 speex/32000
a=rtpmap: 96 telephone-event/8000
a=fmtp: 96 0-16
a=rtpmap: 0 PCMU/8000
a=rtpmap: 3 GSM/8000
a=rtpmap: 18 G729/8000
a=sendrecv
a=rtcp-mux
a=crypto: 1 AES_256_CM_HMAC_SHA1_80 inline:KfYa2ghev8Bz8owzx7FXz7dwQao7K7eSk+leKwrQ5gMBYCQIYI7DBJx0hzSg
a=crypto: 2 AES_CM_128_HMAC_SHA1_80 inline:ZKXoLG8wLD7y1nTtbL1rK0la8897j7iQZ9EymX72
a=crypto: 3 AES_CM_128_HMAC_SHA1_32 inline:djPcgQ5SPn0X9DtFq7KlwicVE0bbRpamj1KbTa7H
```

Answer from Microsoft Direct Routing:

```
m=audio 50918 RTP/SAVP 101 0 18
c=IN IP4 3.3.3.3
a=rtcp: 50918
a=rtcp-mux
a=label: main-audio
a=mid: 1
a=crypto: 2 AES_CM_128_HMAC_SHA1_80 inline:AloudG2m2xliy9SfXeJz6Ac9srDyJUEXTzSEbRCa|2^31
a=sendrecv
a=rtpmap: 101 telephone-event/8000
a=fmtp: 101 0-16
a=rtpmap: 0 PCMU/8000
a=rtpmap: 18 G729/8000
a=fmtp: 18 annexb=no
a=ptime: 20
```

Answer to SIP UA:

```
m=audio 10106 RTP/AVP 101 0 18 3 97
c=IN IP4 2.2.2.2
a=rtpmap: 101 telephone-event/8000
a=fmtp: 101 0-16
a=rtpmap: 0 PCMU/8000
a=rtpmap: 18 G729/8000
a=fmtp: 18 annexb=no
a=rtpmap: 3 GSM/8000
a=rtpmap: 97 speex/8000
a=sendrecv
a=label: main-audio
a=mid: 1
a=ptime: 20
```

Transcoding on outbound call leg

Transcoding activation for outbound calls still needs to be activated in A rules of the originating call agent or realm. This makes the decision when to use transcoding and the appropriate rule conditions more tricky.

For example, if transcoding needs to be activated for outbound calls towards SIP/PSTN users, there needs to be a rule added into A rules of the ms realm.

In following example transcoding is activated for a single number and allows transcoding between PCMU/8000, GSM/8000, G729/8000 and speex/8000 codecs.

Conditions			
Match on:	Operator:	Value:	Description:
<div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;">R-URI User</div> <div style="border: 1px solid #ccc; padding: 2px; width: 100px; float: right;">Add condition</div>	==	<div style="border: 1px solid #ccc; padding: 2px;">+420732541104</div>	✖ If username in request URI...
Actions			
Action:	Value:	Description:	
<div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;">Activate transcoding</div>	<div style="border: 1px solid #ccc; padding: 2px;">PCMU/8000,GSM/8000,G729/8000,speex/8000</div>	<div style="border: 1px solid #ccc; padding: 2px; width: 100px; float: right;">↓ ✖ Activate transcoding for list of codecs.</div>	

Figure 25: Transcoding from Microsoft Direct Routing side

The offer from Microsoft Direct Routing may look like:

```
m=audio 49876 RTP/SAVP 104 9 103 111 18 0 8 97 101 13 118
c=IN IP4 3.3.3.3
a=rtcp: 49877
a=ice-ufrag: /y6p
a=ice-pwd: 0hsTW9ke6A4CJWb8vqEBWFNv
a=rtcp-mux
a=candidate: 1 1 UDP 2130706431 3.3.3.3 49876 typ srflx raddr 10.0.136.32 rport 49876
a=candidate: 1 2 UDP 2130705918 3.3.3.3 49877 typ srflx raddr 10.0.136.32 rport 49877
a=candidate: 2 1 tcp-act 2121006078 3.3.3.3 49152 typ srflx raddr 10.0.136.32 rport 49152
a=candidate: 2 2 tcp-act 2121006078 3.3.3.3 49152 typ srflx raddr 10.0.136.32 rport 49152
a=label: main-audio
a=mid: 1
a=crypto: 1 AES_CM_128_HMAC_SHA1_80 inline:Lv10K2AaxNGl4siPX2x/pGUOX3IVoLu/wzPltT2g+|2^31
a=sendrecv
a=rtpmap: 104 SILK/16000
a=rtpmap: 9 G722/8000
a=rtpmap: 103 SILK/8000
a=rtpmap: 111 SIREN/16000
a=fmtp: 111 bitrate=16000
a=rtpmap: 18 G729/8000
a=fmtp: 18 annexb=no
a=rtpmap: 0 PCMU/8000
a=rtpmap: 8 PCMA/8000
a=rtpmap: 97 RED/8000
a=rtpmap: 101 telephone-event/8000
a=fmtp: 101 0-16
a=rtpmap: 13 CN/8000
a=rtpmap: 118 CN/16000
a=ptime: 20
```

The offer forwarded towards SIP UA for which the transcoding gets activated would be like:

```
m=audio 10478 RTP/AVP 104 9 103 111 18 0 8 97 101 13 118 3 119
c=IN IP4 2.2.2.2
a=rtpmap: 104 SILK/16000
a=rtpmap: 9 G722/8000
a=rtpmap: 103 SILK/8000
```

```
a=rtpmap: 111 SIREN/16000
a=fmtp: 111 bitrate=16000
a=rtpmap: 18 G729/8000
a=fmtp: 18 annexb=no
a=rtpmap: 0 PCMU/8000
a=rtpmap: 8 PCMA/8000
a=rtpmap: 97 RED/8000
a=rtpmap: 101 telephone-event/8000
a=fmtp: 101 0-16
a=rtpmap: 13 CN/8000
a=rtpmap: 118 CN/16000
a=rtpmap: 3 GSM/8000
a=rtpmap: 119 speex/8000
a=sendrecv
a=label: main-audio
a=mid: 1
a=ptime: 20
```

Answer from the SIP UA; in this case it supports just speex codec:

```
m=audio 4000 RTP/AVP 101 119
c=IN IP4 1.1.1.1
b=TIAS: 44000
a=rtcp: 4001 IN IP4 1.1.1.1
a=sendrecv
a=rtpmap: 101 telephone-event/8000
a=fmtp: 101 0-16
a=rtpmap: 119 speex/8000
```

Answer forwarded towards Microsoft Direct Routing contains the codecs allowed for transcoding again:

```
m=audio 10524 RTP/SAVP 101 119 0 3 18
c=IN IP4 1.1.1.1
b=TIAS: 44000
a=rtpmap: 101 telephone-event/8000
a=fmtp: 101 0-16
a=rtpmap: 119 speex/8000
a=rtpmap: 0 PCMU/8000
a=rtpmap: 3 GSM/8000
a=rtpmap: 18 G729/8000
a=sendrecv
a=rtcp-mux
a=crypto: 1 AES_CM_128_HMAC_SHA1_80 inline:n24wNyCfOFISE5PyR6hV8qqntGN7PLJoiIO01bNt
```

Troubleshooting

Monitoring pages

Frafos ABC SBC offers several monitoring pages that can be used for checking proper behavior in appropriate areas. They are available through Monitoring menu of cluster configuration master.

Frafos ABC monitor

Connecting the SBC to Frafos ABC Monitor will significantly help to identify possible issues and to investigate them. In monitor GUI the administrator can use several dashboards focused on specific aspects of the SBCs in cluster.

Traffic logging

One of basic troubleshooting tools of the SBC is “Log received traffic” action that allows to log SIP or SIP and RTP traffic into a pcap file that can be either opened by wireshark or displayed by Frafos ABC monitor.

Significant advantage of traffic logging compared to manually captured PCAPs is, that signaling messages are stored as plain text even if they were sent encrypted using TLS.

Please note that RTP logging may cause quite high resource consumption and should be used with care, especially in production environments.

Traffic logging should be activated in A rules of the appropriate realm or call agent and can look like following example.

Action:	Value:	Description:
Log received traffic		✕ Log SIP/RTP traffic into PCAP file.
Log type	<div>SIP and RTP</div>	
PCAP file name	<div></div>	
Event attributes	<div></div>	

Figure 26: Traffic logging

SEMS log

SEMS process log is another important information source for troubleshooting. The log file is stored in `/var/log/frafos/sems.log` or can be seen in journal via `journalctl` command.

Decrypt TLS traffic with wireshark

In global configuration set option “Dump TLS session keys to file”. In this case SEMS process logs TLS keys used for signaling traffic into a file (`/data/pcap/tls_keys/signaling`) which is readable by wireshark.

In wireshark set TLS protocol preference “(Pre)-Master-Secret log filename” to point to this file and open appropriate PCAP. Note that it might be necessary to capture whole TLS connection including its establishment in the PCAP file to be opened.

References

- [Microsoft Direct Routing documentation](#)
- [Plan Microsoft Direct Routing](#)
- [Teams troubleshooting: Issues that affect call transfers](#)
- [Microsoft 365 admin center](#)
- [Microsoft Teams admin center](#)