ETSI TS 103 698 V1.1.1 (2020-12)

Emergency Communications (EMTEL);
Lightweight Messaging Protocol for
Emergency Service Accessibility (LMPE)

Reference
DTS/EMTEL-00050

Keywords
chat, decentralized identifier, emergency services, location, SSL/TLS certificates

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

This Technical Specification (TS) has been produced by ETSI Special Committee Emergency Communications (EMTEL).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Executive summary

Lightweight Messaging Protocol for Emergency Service accessibility (LMPE) extends a SIP SIMPLE based messaging service with session mode and facilities to redirect or transfer a chat. The mechanisms introduced in the present document differ from existing solutions like MSRP in a sense that no media plane is required. This reduces the functionality to chat, but requires less deployment effort and complexity (e.g. no intermediate services or relays in case of NAT), especially in a roaming use case. In addition, to further reduce complexity, the identification of a user is carried out via a device identifier only, such as a mobile phone number as with comparable chat services. In summary, it simplifies the implementation and thus can be used in simple mobile applications or even smart IoT devices and chatbots, which for example send or respond to messages automatically.

The referred baseline specification (ETSI TS 103 479 [1]) already defines page mode messaging suitable for a single message exchange or a series of short messages similar to paging or SMS on a mobile device. Routing and mapping mechanisms (defined in ETSI TS 103 479 [1]) to determine the proper control room, are based on location information. Therefore a single message exchange is not practicable as caller location may change and lead to messages being routed to a different control room. The present document defines specific message types to group messages into sessions with routing and mapping only required at setup time. In addition the same principles are used to support supplementary services like chat redirect and transfer. Each mechanism is transparent to ETSI TS 103 479 [1] core services and requires only minor modifications to the PSAP interface.

Introduction

Emergency communications services are primarily voice only, along with a marginal share of data and multimedia used by Public Safety Answering Points (PSAPs). Improving access to emergency services for citizens, especially for the deaf and hard of hearing, requires PSAPs and people in need to handle new modes of communications such as text. Messenger services are widespread and well known to the public and currently, the present document defines extension to support a comparable messenger service to access emergency control rooms by leveraging the new architecture introduced in ETSI TS 103 479 [1]. The main purpose of the extensions is to enable a simplified chat session mode combined with means to redirect or transfer a chat session. Furthermore the specification allows a lightweight implementation of a messenger application for emergency chat or bot services. The fact that besides a signalling plane, no further media sessions are required, supports a straight integration with firewalls or, in general, network security technologies.

1 Scope

The purpose of the present document is to describe a lightweight session based emergency chat protocol that extends the base messaging functionality as defined in ETSI TS 103 479 [1]. The messaging service is based only on methods of the SIP signalling plane and interworks with Border Control Function, Emergency Service Routing Proxy, Emergency Call Routing Function, Public Safety Answering Point, the Location Information Server. It is important to emphasize that the introduced feature is an alternative to MSRP, real-time text or, in general, total conversation and not a replacement.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference.

The following referenced documents are necessary for the application of the present document.

[1] ETSI TS 103 479: "Emergency Communications (EMTEL); Core elements for network independent access to emergency services".

[2] IETF RFC 2046: "Multipurpose Internet Mail Extensions (MIME) Part Two: MediaTypes ", Freed N. and Borenstein, N., November 1996.

[3] IETF RFC 3261: "SIP: Session Initiation Protocol", Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and Schooler, E. June 2002.

[4] IETF RFC 3325: "Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity Within Trusted Networks", Jennings, C., Peterson, J. and Watson, M., November 2002.

[5] IETF RFC 3326: "The Reason Header Field for the Session Initiation Protocol (SIP)", Oran, D. and Camarillo, G., December 2002.

[6] IETF RFC 3428: "Session Initiation Protocol (SIP) Extension for Instant Messaging", Campbell, B., Rosenberg, J., Schulzrinne, H., Huitema, C. and Gurle, D., December 2002.

[7] IETF RFC 3841: " Caller Preferences for the Session Initiation Protocol (SIP)"," Rosenberg, J., Schulzrinne, H. and Kyzivat, P., August 2004.

[8] IETF RFC 4119: "A Presence-Based GEOPRIV Location Object Format", Peterson, J., December 2005.

[9] IETF RFC 4244: "An Extension to the Session Initiation Protocol (SIP) for Request History Information", Barnes, M., November 2005.

[10] IETF RFC 4412: "Communications Resource Priority for the Session Initiation Protocol (SIP)", Schulzrinne, H. and Polk, J., February 2006.

[11] IETF RFC 4566: "SDP: Session Description Protocol", Handley, M., Jacobson, V. and Perkins, C., July 2006.

[12] IETF RFC 5031: "A Uniform Resource Name (URN) for Emergency and Other Well-Known Services", Schulzrinne, H., January 2008.

[13] IETF RFC 5621: "Message Body Handling in the Session Initiation Protocol (SIP)", Camarillo, G., September 2009.

[14] IETF RFC 6442: "Location Conveyance for the Session Initiation Protocol", Polk, J., Rosen, B. and Peterson, J., December 2011.

[15] IETF RFC 6881: "Best Current Practice for Communications Services in Support of Emergency Calling", Rosen, B. and Polk, J. March 2013.

[16] IETF RFC 7135: "Registering a SIP Resource Priority Header Field Namespace for Local Emergency Communications", Polk, J. May 2014.

[17] IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3", V Rescorla, E., August 2018.

[18] W3C: "Decentralized Identifiers (DIDs) v1.0 Core Data Model and representations", Working Draft 08 November 2020.

NOTE: Available at https://www.w3.org/TR/did-core/.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents are not necessary for the application of the present document, but they assist the user with regard to a particular subject area.

[i.1] EENA, Version 1.1, March 2013: "Next Generation 112 Long Term Definition".

NOTE: Available at https://eena.org/knowledge-hub/documents/ng112-long-term-definition-standard-for-emergency-services/.

[i.2] EENA Version 1.05, March 2016: "Public Safety Digital Transformation The Internet of Things (IoT) and Emergency Services".

NOTE: Available at https://eena.org/document/the-internet-of-things-and-emergency-services/.

[i.3] W3C Recommendation, November 2019: "Verifiable Credentials Data Model 1.0".

NOTE: Available at https://www.w3.org/TR/vc-data-model/#ecosystem-overview.

3 Definition of terms, symbols and abbreviations

3.1 Terms

Void.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AP	Application Provider
BCF	Border Control Function
BGP	Border Gateway Protocol
CA	Certification Authority
CAP	Common Alerting Protocol
CERT	Computer Emergency Response Team
CR	Carriage Return
DID	Decentralised Identifier
DLT	Distributed Ledger Technology
ECRF	Emergency Call Routing Function
ESInet	Emergency Services IP network
ESRF	Emergency Service Routing Function
ESRP	Emergency Service Routing Proxy
ETSI	European Telecommunications Standards Institute
FG	Forest Guide
GIS	Geographic Information System
HELD	HTTP Enabled Location Delivery
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IETF	Internet Engineering Task Force
IF	Interface
IM	Instant Messaging
IoT	Internet of Things
IP	Internet Protocol
IT	Information Technology
JSON	JavaScript® Object Notation
LF	Line Feed
LIS	Location Information Server
LMPE	Lightweight Messaging Protocol for Emergency Service accessibility
LO	Location Object
LOST	Location to Service Translation
LTD	Long-term Definition
MSD	Minimum Set of Data
MSRP	Message Session Relay Protocol
NAT	Network Address Translation
P-A-I	P-Asserted-Identity
PIDF	Presence Information Data Format
PIDF-LO	Presence Information Data Format - Location Object
PNNS	Protocol Naming and Numbering Service
PSAP	Public Safety Answering Point
RCS	Rich Communication Services
RFC	Request For Comment
SDP	Session Description Protocol
SIP	Session Initiation Protocol
SMS	Short Message Service
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TR	(ETSI) Technical Report
TS	(ETSI) Technical Specification
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
URN	Uniform Resource Name
UTF	Unicode Transformation Format
WGS84	World Geodetic System 1984

4 General

4.1 Overview

Per ETSI TS 103 479 [1] emergency calls are routed by the ESRF to the ESRP via a BCF. Depending on national PSAP models the ESRP may then forward directly to the appropriate PSAP as explained in NG112 LTD [i.1] . The same mechanism applies to instant messaging in a non-session mode. The present document defines certain extensions to interfaces and introduces a mobile application (APP) interface to support a session based chat application. Figure 1 illustrates a high level functional architecture, where specific Application Provider (AP) services are used to manage the application (AP BE) or to interconnect with an ESInet (SIP PROXY). Chat messages addressed to public emergency service SIP URI or service URN are forwarded to a BCF and routed within the ESInet utilizing a geodetic location determined by the mobile application (typically via sensor fusion).

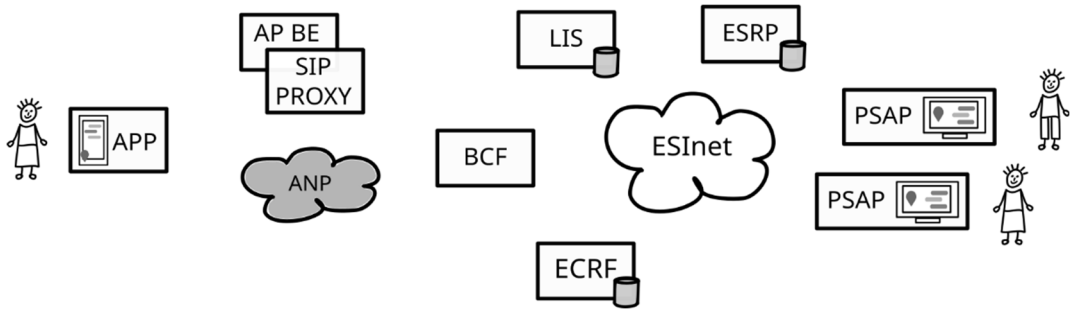

Figure 1: High level functional architecture

The present document specifies only the signalling interface between APP, PSAP and other core services required to setup a session based chat. The following architecture introduces functional elements that comprise an IP only environment. Such elements provide security measures (BCF), emergency call routing (ESRP), mapping PSAP boundaries to SIP URIs (ECRF), a mobile application (APP) and chat processing equipment (PSAP).

4.2 Architecture

The definition of core elements and interfaces supporting a Lightweight Messaging Protocol for Emergency Service accessibility (LMPE) is based on the core concept introduced in ETSI TS 103 479 [1]. LMPE utilizes IP technology and requires public and private managed, and routed IP networks. The present document introduces new interfaces between the functional elements APP and PSAP (dashed-line boxes in Figure 2), and refers to functional elements with their internal and external interfaces as defined in ETSI TS 103 479 [1]; listed below:

- Border Control Function (BCF);
- Emergency Call Routing Function (ECRF);
- Chat Application (APP);
- Public Safety Answering Point (PSAP);
- Emergency Services Routing Proxy (ESRP); and
- Location Information Service (LIS).

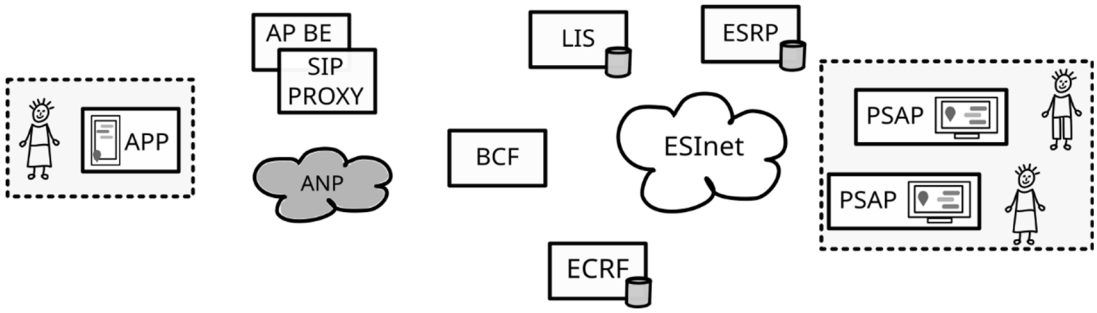

Figure 2: Core elements

4.3 Mandatory Interfaces

Mandatory interfaces are either referenced (ETSI TS 103 479 [1]), or introduced by the present document to define simple chat capabilities of an APP and ESInet core elements. Figure 3 shows interfaces as listed in the following:

- **SIP-1, SIP-2:** Interface between APP, BCF, ESRP and PSAP elements that defines SIP transport and signalling capabilities.
- **LOST-1, LOST-2:** Interface between ESRP and ECRF elements that defines LoST signalling capabilities.
- **HELD-1, HELD-2:** Interface between ESRP or PSAP and LIS elements that defines location dereference and HELD signalling capabilities.
- **IM-2:** APP and PSAP chat handling capabilities to support instant messaging.

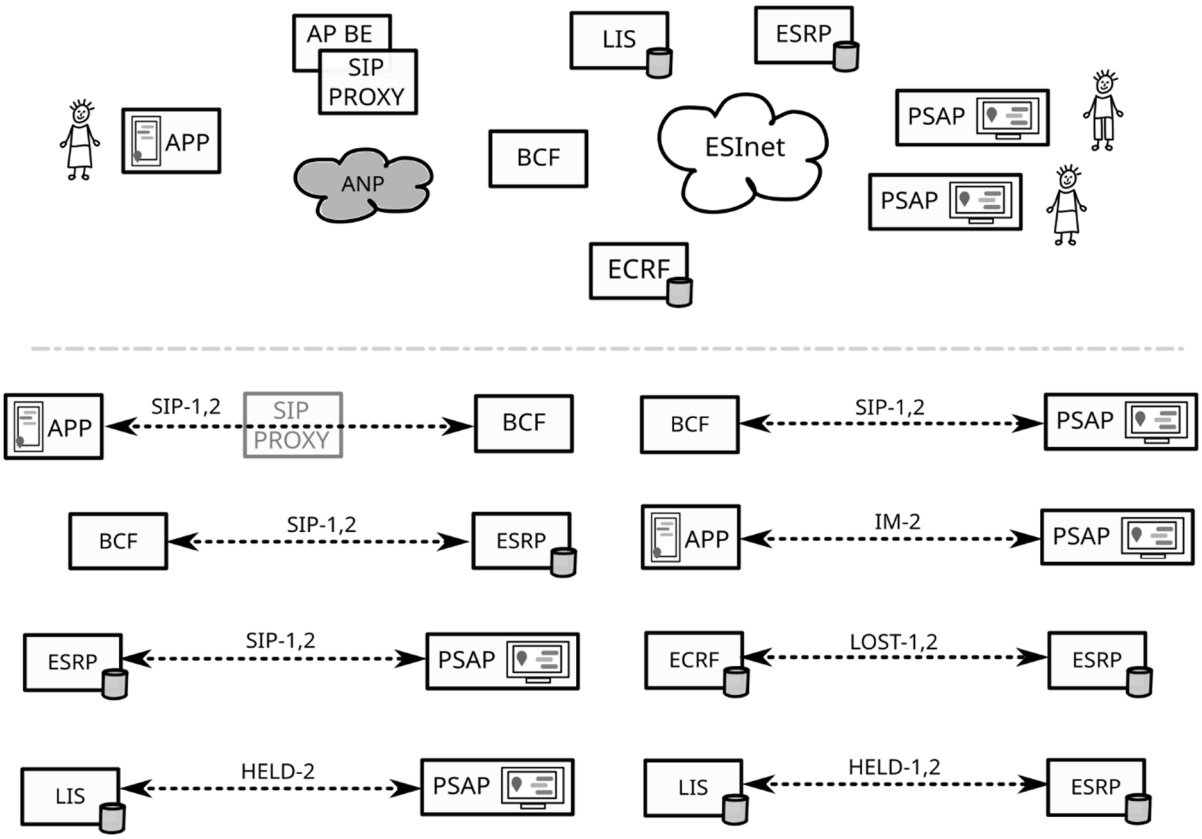

Figure 3: Considered mandatory interfaces

4.4 Optional Interfaces

In addition, optional interfaces as defined in to ETSI TS 103 479 [1], are referenced by the current document to extend mandatory capabilities. Figure 4 shows interfaces as listed in the following:

- **HTTP-2:** Interface between BCF and PSAP elements that defines domain specific web service capabilities.
- **HTTP-3:** Interface between PSAP elements that defines domain specific web service capabilities.
- **LOST-1:** Interface between APP and ECRF elements that defines LoST signalling capabilities.
- **HELD-1:** Interface between APP and LIS elements that defines HELD signalling capabilities.

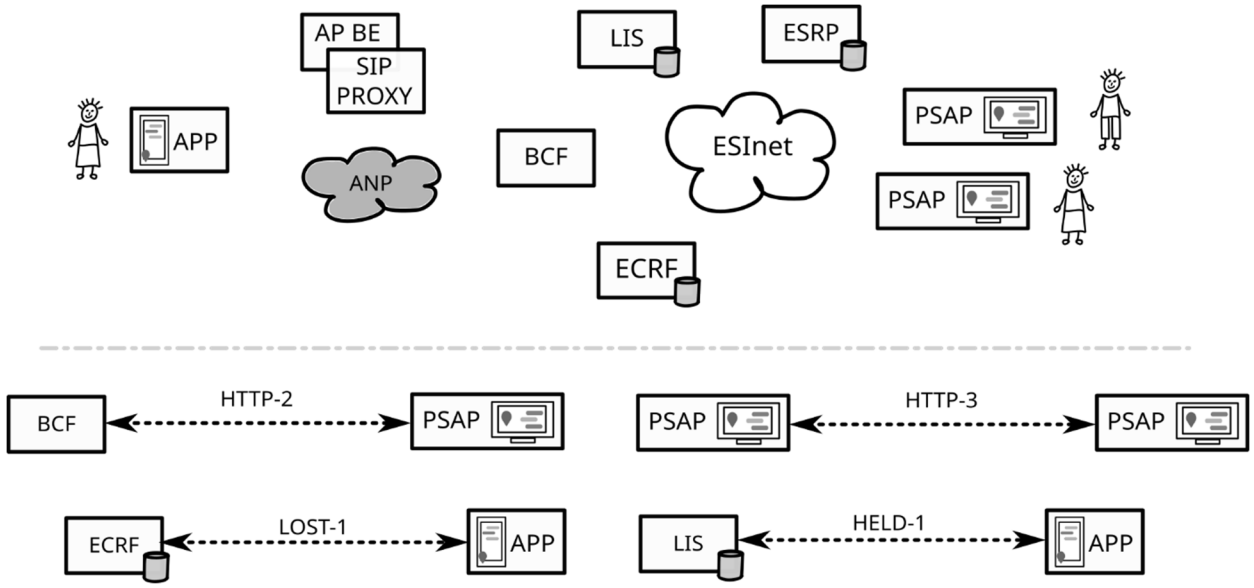

Figure 4: Considered optional interfaces

5 Entities

5.1 Border Control Function (BCF)

5.1.1 Overview

A BCF is the entry point (point-of-interconnect) to the ESInet infrastructure where all traffic from external networks transits. General procedures and interfaces are specified in ETSI TS 103 479 [1].

5.1.2 Mandatory Interfaces

To be compliant with the procedures in the present document, a BCF shall support:

1) the SIP-1 interface as specified in ETSI TS 103 479 [1], clause 6.1.1;
2) the SIP-2 interface as specified in ETSI TS 103 479 [1], clause 6.1.2.

Figure 5 shows mandatory interfaces and neighbouring entities.

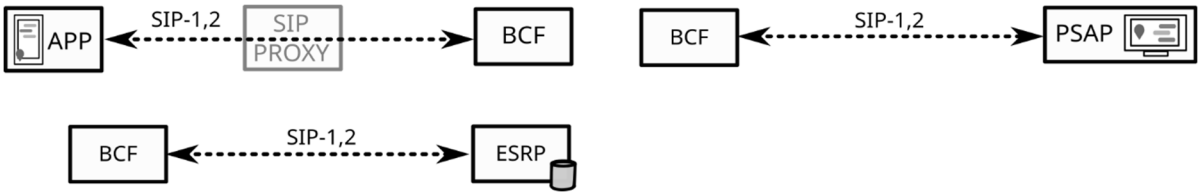

Figure 5: BCF mandatory interfaces

5.1.3 Optional Interfaces

In addition to all mandatory interfaces, a BCF may support any other specific interface as listed in ETSI TS 103 479 [1]:

1) the HTTP-2 interface as specified in ETSI TS 103 479 [1], clause 6.2.2;

Figure 6 shows optional interfaces and neighbouring entities.

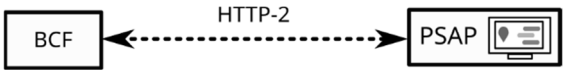

Figure 6: BCF optional interfaces

5.2 Emergency Service Routing Proxy (ESRP)

5.2.1 Overview

The Emergency Service Routing Proxy (ESRP) is the base routing function for emergency chat. General procedures and interfaces are specified in ETSI TS 103 479 [1].

5.2.2 Mandatory Interfaces

To be compliant with the procedures in the present document, an ESRP shall support:

1) the SIP-1 interface as specified in ETSI TS 103 479 [1], clause 6.1.1;
2) the SIP-2 interface as specified in ETSI TS 103 479 [1], clause 6.1.2;
3) the LOST-1 interface as specified in ETSI TS 103 479 [1], clause 6.4.1;
4) the LOST-2 interface as specified in ETSI TS 103 479 [1], clause 6.4.2;
5) the HELD-1 interface as specified in ETSI TS 103 479 [1], clause 6.5.1;
6) the HELD-2 interface as specified in ETSI TS 103 479 [1], clause 6.5.2.

Figure 7 shows mandatory interfaces and neighbouring entities.

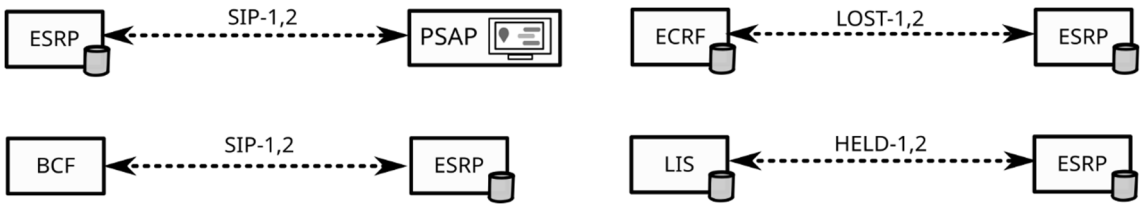

Figure 7: ESRP mandatory interfaces

5.2.3 Optional Interfaces

In addition to all mandatory interfaces, an ESRP may support any other specific interface as listed in ETSI TS 103 479 [1].

5.3 Emergency Call Routing Function (ECRF)

5.3.1 Overview

The Emergency Call Routing Function (ECRF) is the base mapping function for emergency chat. General procedures and interfaces are specified in ETSI TS 103 479 [1].

5.3.2 Mandatory Interfaces

To be compliant with the procedures in the present document, an ECRF shall support:

1) the LOST-1 interface as specified in ETSI TS 103 479 [1], clause 6.4.1;

2) the LOST-2 interface as specified in ETSI TS 103 479 [1], clause 6.4.2.

Figure 8 shows mandatory interfaces and neighbouring entities.

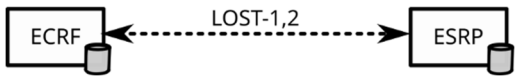

Figure 8: ECRF mandatory interfaces

5.3.3 Optional Interfaces

In addition to all mandatory interfaces, an ECRF may support any other specific interface as listed in ETSI TS 103 479 [1].

5.4 Public Safety Answering Point (PSAP)

5.4.1 Overview

A PSAP is a service, typically composed of more than one functional element. The functional elements that make up a PSAP are out of scope of the present document. The PSAP implements the LMPE interface including the chat capability as specified in the present document.

5.4.2 Mandatory Interfaces

To be compliant with the procedures in the present document, a PSAP shall support:

1) the SIP-1 interface as specified in ETSI TS 103 479 [1], clause 6.1.1;

2) the SIP-2 interface as specified in ETSI TS 103 479 [1], clause 6.1.2;

3) the HELD-2 interface as specified in ETSI TS 103 479 [1], clause 6.5.2;

4) the IM-2 interface as specified in clause 6.2.

Figure 9 shows mandatory interfaces and neighbouring entities.

Figure 9: PSAP mandatory interfaces

5.4.3 Optional Interfaces

In addition to all mandatory interfaces, a PSAP may support any other specific interface as listed in ETSI TS 103 479 [1] and below.

1) the HTTP-2 interface as specified in ETSI TS 103 479 [1], clause 6.2.2;

2) the HTTP-3 interface as specified in clause 6.3.

Figure 10 shows optional interfaces and neighbouring entities.

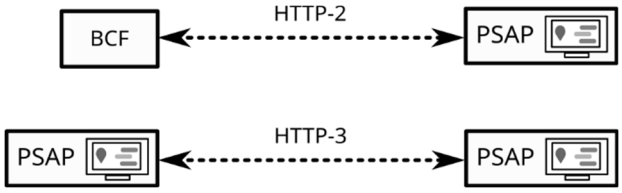

Figure 10: PSAP optional interfaces

5.5 Location Information Server (LIS)

5.5.1 Overview

Location is fundamental to the operation of the emergency services, and the generic functional entity that provides location is a Location Information Server (LIS) as specified in ETSI TS 103 479 [1].

5.5.2 Mandatory Interfaces

To be compliant with the procedures in the present document, a LIS shall support:

1) the HELD-1 interface as specified in ETSI TS 103 479 [1], clause 6.5.1;

2) the HELD-2 interface as specified in ETSI TS 103 479 [1], clause 6.5.2.

Figure 11 shows mandatory interfaces and neighbouring entities.

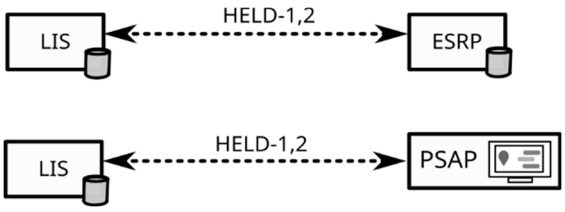

Figure 11: LIS mandatory interfaces

5.5.3 Optional Interfaces

In addition to all mandatory interfaces, a LIS may support any other specific interface as listed in ETSI TS 103 479 [1].

5.5.4 Location Representation

Location is represented by content in a PIDF-LO document (IETF RFC 4119 [8]). All geodetic data shall use WGS84 as the datum. The representation of the location object within the PIDF document shall utilize the `tuple` element as defined in IETF RFC 4119 [8].

5.6 Chat Application (APP)

5.6.1 Overview

The APP implements the LMPE interface including the chat capability as specified in the present document. Other backend services required by the APP are out of scope of the present document.

5.6.2 Mandatory Interfaces

To be compliant with the procedures in the present document, an APP shall support:

1) the SIP-1 interface as specified in ETSI TS 103 479 [1], clause 6.1.1;

2) the SIP-2 interface as specified in ETSI TS 103 479 [1], clause 6.1.2;

3) the IM-2 interface as specified in clause 6.2.

Figure 12 shows mandatory interfaces and neighbouring entities.

Figure 12: LIS mandatory interfaces

5.6.3 Optional Interfaces

In addition to all mandatory interfaces, an APP may support any other specific interface as listed in ETSI TS 103 479 [1] and below.

1) the LOST-1 interface as specified in ETSI TS 103 479 [1], clause 6.4.1;

2) the HELD-1 interface as specified in ETSI TS 103 479 [1], clause 6.5.1.

Figure 13 shows optional interfaces and neighbouring entities.

Figure 13: APP optional interfaces

5.6.4 Location Representation

Location is represented by content in a PIDF-LO document (IETF RFC 4119 [8]). All geodetic data shall use WGS84 as the datum. The representation of the location object within the PIDF document shall utilize the `tuple` element as defined in IETF RFC 4119 [8].

6 Interfaces

6.1 Signalling

6.1.1 SIP Transport (SIP-1)

SIP signalling within the ESInet shall be TCP with TLS as defined in IETF RFC 8446 [17]. When a TLS connection already exists, either peering entity shall reuse that TLS connection for all SIP messages within a chat (refer to clause 6.2.3) by utilizing a proper session timeout of at least 3 minutes. Fallback to UDP is allowed. However emergency call messages have many large elements, for example a PIDF-LO, and are more likely to be fragmented when carried in UDP. Fragmentation and reassembly shall be supported by all ESInet elements. If TLS establishment fails, fallback to transport protocols without TLS is allowed.

If fallback with TLS occurs, additional security weaknesses should be considered, and implementations should be prepared to deal with the security risks when TLS protection is not available. Known attacks on incomplete fragmentation/reassembly implementations are another concern, which should be addressed by all elements in the ESInet. Persistent TLS connections between elements that frequently exchange SIP transactions should be deployed.

TLS implementations shall support mutual authentication (using at least RSA-1024), which implies both ends have an X.509 certificate available to the other party. How a certificate is created and issued by a Certificate Authority (CA) is out of scope of the present document.

6.1.2 SIP Session (SIP-2)

6.1.2.1 Overview

The call interface is SIP (as defined in IETF RFC 3261 [3]). All chat messages presented to the ESInet shall be SIP signalled as specified in ETSI TS 103 479 [1], clause 6.7.

6.1.2.2 SIP Methods

MESSAGE:

The MESSAGE method, an extension to SIP, allows the transfer of Instant Messages and is also used to carry a Common Alerting Protocol (CAP) message. Since the MESSAGE request is an extension to SIP, it inherits all the request routing and security features of that protocol. MESSAGE requests carry the content in the form of MIME body parts. MESSAGE requests do not themselves initiate a SIP dialog or session.

MESSAGE requests may be sent in the context of a dialog or session initiated by some other SIP request (such as INVITE), for example in a multi-media call or text messaging session. For more information on MESSAGE please refer to IETF RFC 3428 [6]. Non-human-associated calls are sent using MESSAGE requests outside of a session. Text messages or instant messages may be sent using MESSAGE within a session (in which case an interactive associated stream of such messages is established) or outside a session (in which case a set of disconnected stand-alone messages are sent). MESSAGE is part of the SIP/SIMPLE presence and messaging system.

6.1.2.3 Required SIP Headers

Table 1 shows the SIP header fields required in the MESSAGE methods, recalling that the Request-URI will contain `urn:service:sos` or a sub-service of it as defined in IETF RFC 6881 [15], section 5.

Table 1: Required SIP Headers

Header Field/Request	Defined In	See section (or IETF RFC 6881 [15])	Notes
Request-URI	IETF RFC 3261 [3], section 8.1.1.1	ED62 1.	"urn:service:sos" or a subservice of it
To	IETF RFC 3261 [3], sections 8.1.1.2 & 20.39	ED62 2.	Usually sip:112 or "urn:service:sos"
From	IETF RFC 3261 [3], sections 8.1.1.3 & 20.20	ED62 3.	Content cannot be trusted unless protected by an Identity header
Via	IETF RFC 3261 [3], sections 8.1.1.7 & 20.42		Occurs multiple times, once for each SIP element in the path
CSeq	IETF RFC 3261 [3], sections 8.1.1.5 & 20.16		Defines the order of transactions in a session
Call-ID	IETF RFC 3261 [3], sections 8.1.1.4 & 20.8		This is the SIP call id
Call-Info	IETF RFC 3261 [3], sections 8.1.1.10 & 20.9		May contain Additional Data, Call and Incident Tracking IDs
Content-Length	IETF RFC 3261 [3], section 20.14		
Content-Type	IETF RFC 3261 [3], sections 8.2.3 & 20.15		Used, for example, in IETF RFC 4119 [8] and IETF RFC 4566 [11]
Geolocation	IETF RFC 6442 [14]	ED62 8.	
Geolocation-Routing	IETF RFC 6442 [14]	ED62 8.	Specifies if the Geolocation header field can be used for routing
History-Info	IETF RFC 4244 [9]		Indicates the call has been retargeted
P-Access-Network-Info	IETF RFC 3325 [4]		May contain cell site info in carrier specific formats
P-Asserted-Identity	IETF RFC 3325 [4]		Carries the identity of a device verified by authentication
Reply-To	IETF RFC 3261 [3], section 20.31		Carries the public identity of a PSAP

6.1.2.4 Accepted SIP Headers

Table 2 shows the SIP header fields accepted in SIP methods.

Table 2: Accepted SIP Headers

Header Field	Defined In	Notes
Max-Forwards	IETF RFC 3261 [3], section 20.22	Specifies the maximum number of SIP elements that may be traversed before assuming a routing loop has occurred
Accept-Contact	IETF RFC 3841 [7]	
Accept	IETF RFC 3261 [3], section 20.1	
Content-Encoding	IETF RFC 3261 [3], section 20.12	
Accept-Encoding	IETF RFC 3261 [3], section 20.2	
Content-Language	IETF RFC 3261 [3], section 20.13	
Accept-Language	IETF RFC 3261 [3], section 20.3	
Content-Disposition	IETF RFC 3261 [3], section 20.11	
Allow	IETF RFC 3261 [3], section 20.5	
Unsupported	IETF RFC 3261 [3], section 20.40	
Require	IETF RFC 3261 [3], section 20.32	
Proxy-Require	IETF RFC 3261 [3], section 20.29	
Expires	IETF RFC 3261 [3], section 20.19	
Subject	IETF RFC 3261 [3], section 20.36	
Priority	IETF RFC 3261 [3], section 20.26	
Date	IETF RFC 3261 [3], section 20.17	
Timestamp	IETF RFC 3261 [3], section 20.38	
Organization	IETF RFC 3261 [3], section 20.25	
User-Agent	IETF RFC 3261 [3], section 20.41	
Server	IETF RFC 3261 [3], section 20.35	
Authorization	IETF RFC 3261 [3], section 20.7	
Authentication-Info	IETF RFC 3261 [3], section 20.6	
Proxy-Authenticate	IETF RFC 3261 [3], section 20.27	

Header Field	Defined In	Notes
Proxy-Authorization	IETF RFC 3261 [3], section 20.28	
WWW-Authenticate	IETF RFC 3261 [3], section 20.44	
Warning	IETF RFC 3261 [3], section 20.43	
Error-Info	IETF RFC 3261 [3], section 20.18	
In-Reply-To	IETF RFC 3261 [3], section 20.21	
Retry-After	IETF RFC 3261 [3], section 20.33	
Resource-Priority	IETF RFC 4412 [10], section 3.1	

6.1.2.5 Resource Priority

The resource priority header (as defined in IETF RFC 4412 [10]) is used on SIP calls to indicate priority that proxy servers give to specific calls. All SIP user agents that place calls within the ESInet shall be able to set Resource Priority. All SIP proxy servers in the ESInet shall implement Resource Priority and process calls in priority order when a queue of calls is waiting for service at the proxy server and, where needed, pre-empt lower priority calls.

BCFs shall police Resource Priority for incoming SIP calls. Calls that appear to be emergency calls shall be marked with a provisioned Resource Priority, which defaults to `esnet.1`. PSAP callbacks during handling of an incident use `esnet.0`. Callbacks outside of an incident are not marked. ESInets normally use the esnet namespace (as defined in IETF RFC 7135 [16]). The use of the namespace in an ESInet is defined as shown in Table 3.

Table 3: Resource Priority

esnet.0	calls which relate to an incident in progress, but whose purpose is not critical
esnet.1	emergency calls traversing the ESInet
esnet.2	calls related to an incident in progress which are deemed critical
esnet.3- esnet.7	not defined

6.1.2.6 History-Info and Reason

When a call is retargeted by any routing element, the receiving entity shall have the ability to know why it got the call. For this reason, SIP elements in the ESInet shall support the History-Info header (as defined in IETF RFC 4244 [9]) and the associated Reason header (IETF RFC 3326 [5]). Elements which retarget a call, shall add a History-Info header indicating the original intended recipient, and the reason why the call was retargeted. ESInet elements shall be prepared to handle a History-Info (and its associated Reason header) added by any SIP element.

6.1.2.7 Call-Info

SIP MESSAGE transactions may contain a `Call-Info` header field with a URI referencing one or more Additional Data blocks. The transaction to dereference the Additional Data shall be protected with TLS. The dereferencing entity, which may be a PSAP, uses its credentials to dereference the Additional Data URI and should have means to render information to the user.

The entity initiating a chat shall assign the Call Identifier. The form of a Call Identifier is a URN (see IETF RFC 5031 [12]) formed by the prefix `urn:emergency:service:uid:callid:`, a unique string containing alpha and/or numeric characters, the `":"` character, and the Element Identifier of the element that first handled the call. The unique string portion of the Call Identifier shall be unique for each call the element handles over time. The length of the unique string portion of the Call Identifier shall be between 10 and 30 characters. The Call Identifier is added to a SIP message using the `Call-Info` header field with a purpose of `EmergencyCallData.CallId`. The following example illustrates the use of Call-Info to provide a Call Identifier:

```
Call-Info: <urn:emergency:uid:callid:a56e556d871:dec112.at>;purpose=EmergencyCallData.CallId
```

The entity initiating a chat shall assign the Message Identifier. The form of a Message Identifier is a URN (see IETF RFC 5031 [12]) formed by the prefix `urn:emergency:service:uid:msgid:`, a unique string containing numeric characters, the ":" character, and the Element Identifier of the service that initiates the chat. The unique string portion of the Message Identifier shall be unique for each message the element sends within a chat. The unique string portion of the Message Identifier shall be an integer value starting at 1 and is monotonically increased by one (1) for each new message. The Message Identifier is added to a SIP message using the `Call-Info` header field with a purpose of `EmergencyChatData.MsgId`. The following example illustrates the use of Call-Info to provide a Message Identifier:

```
Call-Info:
<urn:emergency:service:uid:msgid:1:service.dec112.at>;purpose=EmergencyCallData.
MsgId
```

The entity initiating a chat shall assign the Message Type. The form of a Message Type is a URN (see IETF RFC 5031 [12]) formed by the prefix `urn:emergency:service:uid:msgtype:`, a unique string containing numeric characters, the ":" character, and the Element Identifier of the service that initiates the chat. The unique string portion of the Message Type shall be unique for each message the element sends within a chat. The format of the unique string portion of the Message Type shall be a 16-bit integer value. The Message Type is added to a SIP message using the `Call-Info` header field with a purpose of `EmergencyCallData.MsgType`.

Table 4 defines version 1 of the 16-bit encoding used in the present document. The semantic for each message type is given in clause 6.2.

Table 4: Message Types

```
0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-----------+---+---------------+
| reserved  | v |    type       | v...version
+-----------+---+---------------+
|0 0 0 0 0 0|0 1|0 0 0 0 0 0 0 0| unknown
|0 0 0 0 0 0|0 1|0 0 0 0 0 0 0 1| start
|0 0 0 0 0 0|0 1|0 0 0 0 0 0 1 1| in-chat
|0 0 0 0 0 0|0 1|0 0 0 0 0 0 1 0| stop
|0 0 0 0 0 0|0 1|0 0 0 0 0 1 0 0| heartbeat
|0 0 0 0 0 0|0 1|0 0 0 0 1 0 x x| transfer
|0 0 0 0 0 0|0 1|0 0 0 1 0 0 x x| redirect
|0 0 0 0 0 0|0 1|0 0 1 x x x x x| reserved
|0 0 0 0 0 0|0 1|0 1 x x x x x x| reserved
|0 0 0 0 0 0|0 1|1 0 0 0 0 x 0 0| inactive
+-----------+---+---------------+
```

The following example illustrates the use of Call-Info to provide the Message Type of `in-chat` messages sent either from the APP or the PSAP:

```
Call-Info:
<urn:emergency:service:uid:msgtype:259:service.dec112.at>;purpose=EmergencyCallData.MsgType
```

6.1.2.8 SIP Message Bodies

All SIP elements forwarding chat messages shall support multipart MIME types as defined in IETF RFC 2046 [2] and shall support multipart message handling as specified in IETF RFC 5621 [13]. The content type for chat text shall be `text/plain; charset=utf-8`. Location and session description may be present in a message body. All SIP elements shall allow additional body content (for example, images, vcards, eCall MSD, etc.) to pass to the PSAP.

6.1.2.9 SIP Element Overload

Any SIP element may encounter a condition in which it is asked to process more calls than it can handle. Elements shall not return `503 Busy Here` unless it is certain, by design and configuration that the upstream element can reliably cope with the error. To cope with such overload, SIP elements may implement the mechanisms described in ETSI TS 103 479 [1].

6.1.2.10 Test Call

Elements in the SIP signalling path shall implement the test function described in IETF RFC 6881 [15]. As the function is designed to test if an emergency chat was placed from the test-initiating device, the test mechanism should mimic the entire actual path as closely as practical. Further the test mechanism shall be automatic, with no manual intervention required.

A device initiating a test chat (APP) shall follow the procedures as defined in clause 6.2.2 and shall use `urn:service:sos.test` to be interpreted as a request to initiate a test chat. The PSAP should return a `200 OK` response in normal conditions, indicating that it will complete the test function. The PSAP may limit the number of test calls. If that limit is exceeded, the response shall be `486 Busy Here`. PSAPs should accept requests for secondary services such as `urn:service:sos.fire.test` and complete a test call. PSAP management may disable the test function (according to the PSAP policy).

If the PSAP accepts the test, it should return an automatic SIP MESSAGE to immediately close the chat (as defined in clause 6.2.4) with a body with MIME type `text/plain; charset=utf-8` consisting of the following contents:

- The name of the PSAP, terminated by a CR and LF;
- The service URN received, terminated by a CR and LF;
- The location reported with the call (in the geolocation header).

If the location was provided by value, the response would be a natural text version of the received location. If the location was provided by reference, the PSAP should dereference the location, using credentials acceptable to the LIS issued specifically for test purposes. The location returned may not be the same as the LIS would issue for an actual emergency chat. A PSAP should refuse repeated requests for test from the same source in a short period of time (e.g. within 2 minutes). Any refusal is signalled with a `486 Busy Here`.

6.1.2.11 Decentralised Identifier (DID)

The entity initiating a chat may include a Decentralised Identifier (DID), a globally unique identifier that does not require a centralized registration authority because it is registered with Distributed Ledger Technology (DLT) or other form of decentralized network (i.e. verifiable data registry). The form of a DID is defined in Decentralized Identifiers (DIDs) v1.0 Core Data Model and Syntaxes [18]. The DID is added to a SIP message using the `Call-Info` header field with a purpose of `EmergencyCallData.DID`. The following example illustrates the use of Call-Info to provide a DID:

```
Call-Info: <did:example:123456789abcdefghi>;purpose=EmergencyCallData.DID
```

Further details describing workflow and ecosystem of verifiable credentials and the use of DIDs can be found in [i.3]. In a general scenario, the holder (person initiating an emergency chat) submits a credential for verifiers (PSAPs). The credential issued by issuer (authority) is called a verifiable credential. The verifiable credential will include not only an ID (DID format), but also other attributes called claims. As illustrated in Figure 14, the holder of a verifiable credential is mediating between issuer and verifier. The issuer and holder trust each other, the holder trusts the verifier, and the verifier trusts the issuer. Any role in the triangle can be played by a person or institution.

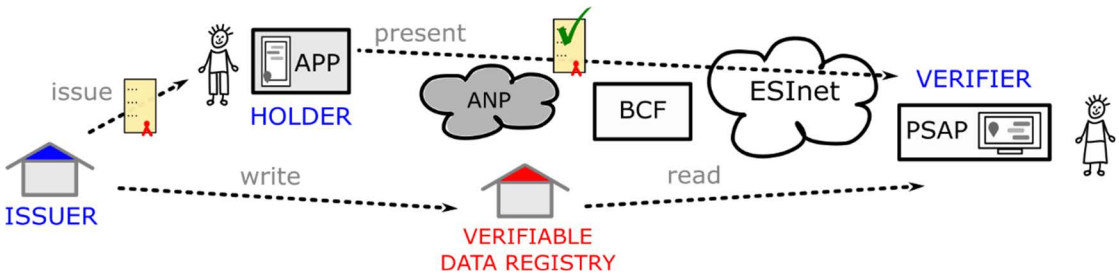

Figure 14: Verifiable credentials triangle of trust

6.2 Instant Messaging (IM-2)

6.2.1 Overview

The Instant Messaging (IM-2) interface supports the creation of a messaging dialog or session mode for messaging applications. Basically, a unique call identifier groups individual messages with a specific message type (e.g. start, stop, in-chat) into a single conversation. This supports chat messages being sent to the same PSAP and additional facilities like chat redirect or chat transfer. The following clauses assume that an APP (emergency messenger application or OS native feature) receives configuration from a backend or via OTA (over-the-air) updates to connect to a national ESInet peering entity (BCF).

6.2.2 Session Mode Initiation

To create a chat session, the originating entity (APP) shall issue an automatic SIP MESSAGE to provide a unique Call Identifier, a Message Identifier and a Message Type set to `start/257`. This first message shall include the selected service in the request URI (either a proper service URN or a service number that addresses an emergency service), location information, as defined in clause 5.6.4, a public routable SIP URI identifying an originating device provided with the From header, and a predefined text (e.g. introducing the person in need). Refer to the message sequence chart in Figure 14.

NOTE 1: The SIP URI provided with the From header can be supplemented by a service provider with a P-A-I header at the SIP Proxy. In the following diagrams only the APP is shown for simplicity and the SIP Proxy is omitted, therefore examples only show the From header.

Upon the reception of the first message, the PSAP shall respond with 200 OK and issue an automatic SIP MESSAGE (refer to clause 6.2.3) to return the received Call Identifier and provide a Message Identifier and a Message Type set to `start/257`. This message shall include a Reply-To header value that provides the public routable SIP URI of the PSAP and a predefined text (e.g. the first question a PSAP may ask in the chat). Refer to the message sequence chart in Figure 15.

NOTE 2: The SIP URI provided with the Reply-To header value may point to the public interface of a BCF. Proper mapping to the internal SIP URI of the receiving PSAP is part of the BCF configuration and out of scope of the present document.

Message Sequence Chart

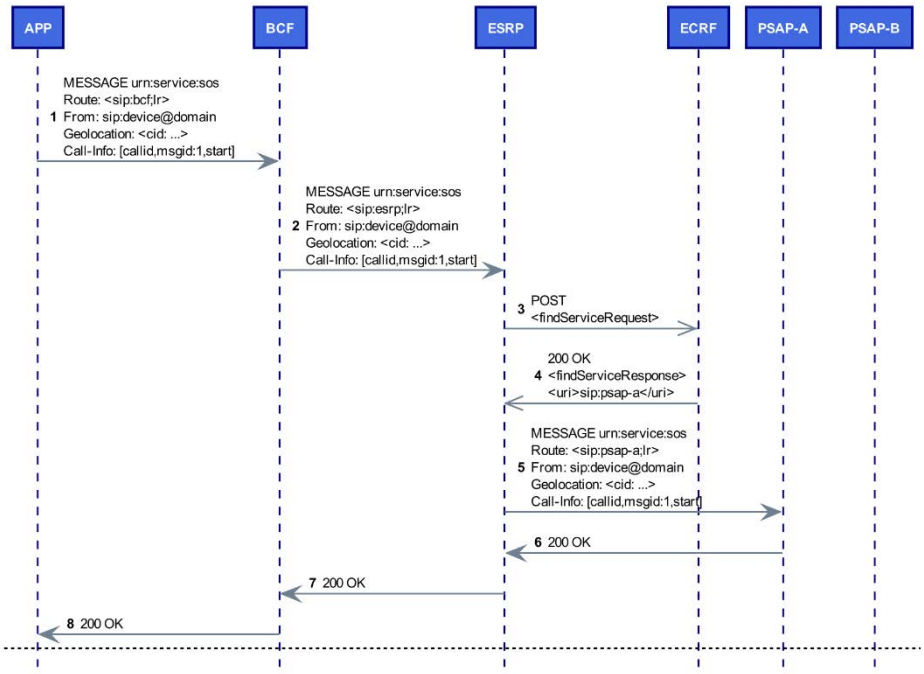

Figure 15: Session Mode Initiation - Part 1

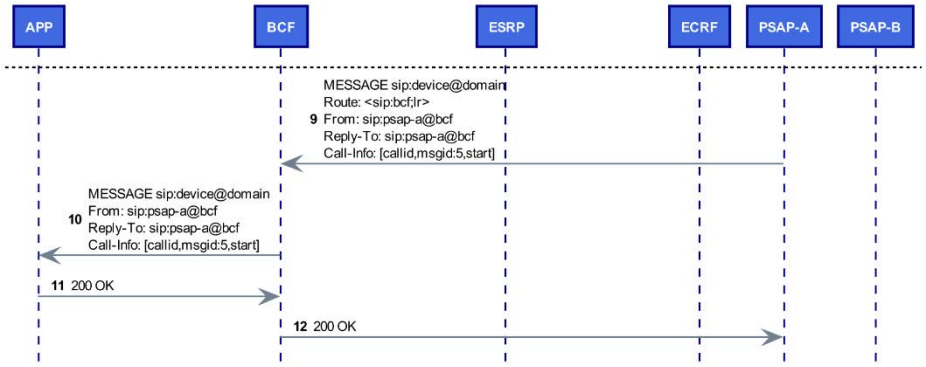

Figure 16: Session Mode Initiation - Part 2

6.2.3 Session Mode Chat

After the reception and acknowledgement of the first PSAP message (`start/257`), either end (APP or PSAP) may continue with SIP MESSAGE transactions in session mode, where the following shall apply:

- Chat messages originating at the APP shall be sent to the public routable SIP URI as received from the PSAP (Reply-To header value) and shall include location information, as defined in clause 5.6.4, the unique Call Identifier, a Message Identifier increased by one (1) for each APP message, and a Message Type set to `in-chat/259`. Refer to the message sequence chart in Figure 16.

- Chat messages originating at the PSAP shall be sent to the public routable SIP URI as received from the APP and identifying an originating device provided with the P-A-I header (in the absence of a P-A-I header value the From header value shall be used). The message shall include the Reply-To header value, a unique Call Identifier, a Message Identifier increased by one (1) for each PSAP message, and a Message Type set to `in-chat/259`. Refer to the message sequence chart in Figure 16.

Message Sequence Chart

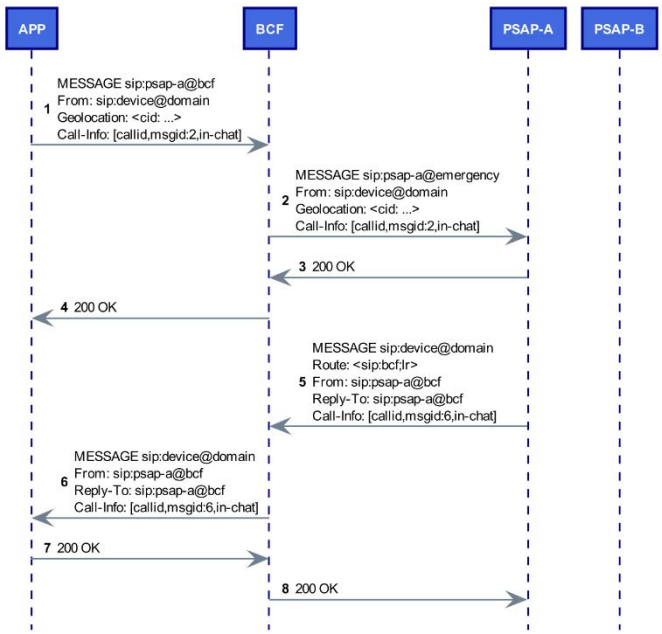

Figure 17: Session Mode Chat

6.2.4 Session Mode Termination

Upon the reception and acknowledgement of the first PSAP message (`start/257`), either end (APP or PSAP) may terminate the session mode with a SIP MESSAGE transaction, where the following shall apply:

- The APP shall send a termination SIP MESSAGE to the public routable SIP URI as received from the PSAP (Reply-To header value) and shall include location information, as defined in clause 5.6.4, the unique Call Identifier, a Message Identifier increased by one (1), and a Message Type set to `stop/258`. The message should also include a predefined text to indicate that the session has been closed by the user. Refer to the message sequence chart in Figure 17.

- The PSAP shall sent a termination SIP MESSAGE to the public routable SIP URI as received from the APP and identifying an originating device provided with the P-A-I header (in the absence of a P-A-I header value the From header value shall be used). The message shall include the Reply-To header value, the unique Call Identifier, a Message Identifier increased by one (1), and a Message Type set to `stop/258`. The message should also include a predefined text to indicate that the session has been closed by the call taker. Refer to the message sequence chart in Figure 17.

Message Sequence Chart

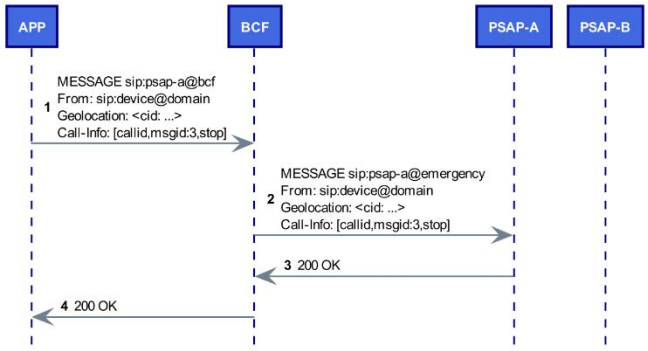

Figure 18: Session Mode Termination

6.2.5 Keep-Alive Messages

Upon the reception and acknowledgement of the first PSAP message (`start/257`), the APP shall and the PSAP may send keep-alive messages with a SIP MESSAGE transaction, where the following shall apply:

- The APP shall periodically send (at least every 20 s) a keep-alive SIP MESSAGE to the public routable SIP URI as received from the PSAP (Reply-To header value) and shall include location information, as defined in clause 5.6.4, the unique Call Identifier, and a Message Type set to `heartbeat/260`.

- The PSAP shall periodically send (at least every 20 s) a keep-alive SIP MESSAGE to the public routable SIP URI as received from the APP and identifying an originating device provided with the P-A-I header (in the absence of a P-A-I header value the From header value shall be used). The message shall include the Reply-To header value, the unique Call Identifier, and a Message Type set to `heartbeat/260`.

NOTE: Keep-alive messages are used to update location information and to keep pinhole/NAT-bindings open.

In the case the APP changes its state to backgrounded or inactive, it may send a keep-alive SIP MESSAGE with the inactive flag set, resulting in a Message Type set to `heartbeat|incative/388`.

6.2.6 Transfer

A PSAP may transfer a chat session to another PSAP any time after the chat session has been established and assuming that transferor and transferee hold an agreement to do so. To negotiate and execute a chat transfer, PSAPs are required to implement the Chat Transfer (HTTP-3) interface as specified in clause 6.3. Refer to clause 6.3 for detailed message timing and user interaction. To initiate a transfer, the following shall apply:

- The PSAP initiating the transfer (message 6 in Figure 19: chatTransferExecutionRequest) shall send a SIP MESSAGE to the public routable SIP URI as received from the APP and identifying an originating device provided with the P-A-I header (in the absence of a P-A-I header value the From header value shall be used). The message shall include the Reply-To header value, the unique Call Identifier, a Message Identifier indicating the last sequence number used by the transferor. The Message Type shall be `stop|transfer/266`, and the message should include a predefined text to indicate that the session will be transferred by the call taker. The PSAP shall not continue sending messages.

- Upon reception of a `stop|transfer/266`, the APP shall respond with a 200 OK and stall user input until it receives a `start|transfer/265` message from the transferee PSAP. Refer to the red block starting at message 8 in Figure 19. The pending transfer shall be visually indicated to the user.

- The transferee PSAP shall then send an automatic SIP MESSAGE to the public routable SIP URI (from header value) and include its Reply-To header value, the unique Call Identifier, the Message Identifier indicating the last sequence number plus one (1) as received via the chatTransferExecutionRequest from the transferor PSAP. The Message Type shall be `start|transfer/265`, and the message should include a predefined text to indicate that the session has been transferred to a new PSAP.

- After the reception and acknowledgement of the first PSAP message (`start|transfer/265`), see message 15 in Figure 20, the APP shall not accept further (`start|transfer/265`) messages until another `stop|transfer/266` is received. Either end (APP or PSAP) may continue with SIP MESSAGE transactions in session mode as defined in clause 6.2.3.

Message Sequence Chart

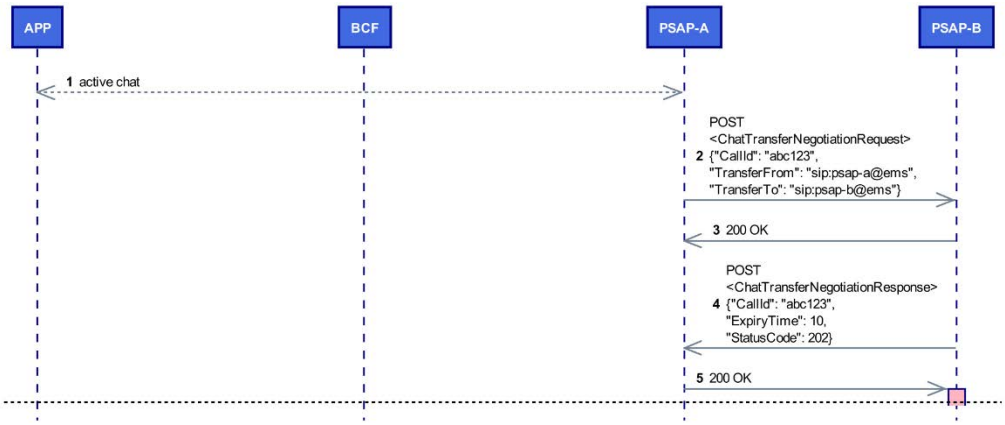

Figure 19: Transfer - Part 1

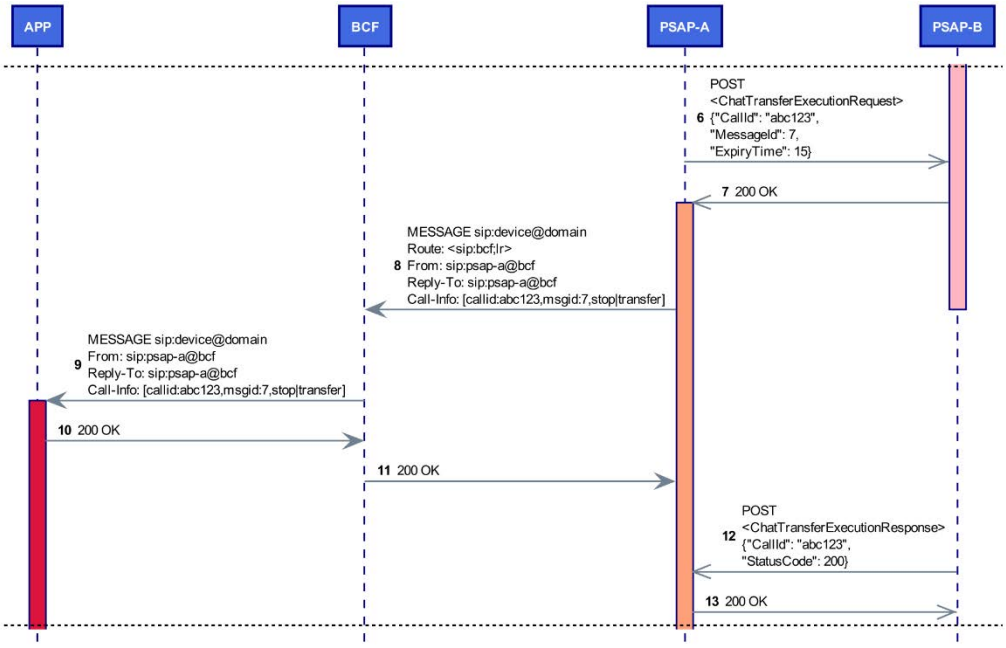

Figure 20: Transfer - Part 2

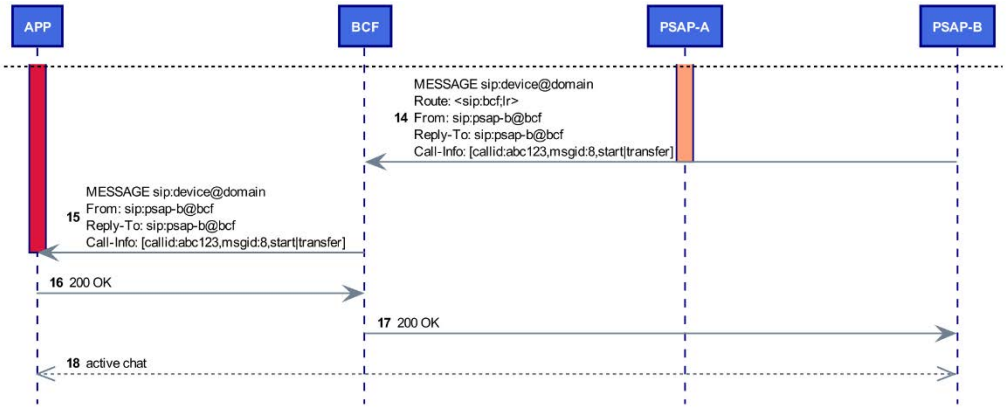

Figure 21: Transfer - Part 3

6.2.7 Redirect

A PSAP may redirect a chat session to another PSAP immediately after the chat session has been established, assuming PSAPs hold an agreement to do so. To initiate a redirect, the following shall apply:

- The PSAP initiating the redirect shall send a SIP MESSAGE to the public routable SIP URI as received from the APP and identifying an originating device provided with the P-A-I header (in the absence of a P-A-I header value the From header value shall be used). The message shall include a Reply-To header value identifying the redirect target, the unique Call Identifier, a Message Identifier indicating the last sequence number used. The message type shall be `stop|redirect/274`, and the message should include a predefined text to indicate that the session will be redirected. The PSAP shall not continue sending messages.

- Upon reception of a `stop|redirect/274`, the APP shall respond with a 200 OK and stall user input until it receives a `start/257` message from the new PSAP. Refer to the red block starting at message 6 in Figure 21. The pending redirection shall be visually indicated to the user.

- The APP shall then send an automatic SIP MESSAGE to the public routable SIP URI (From header value) as received from the redirecting PSAP (via the `stop|redirect/274` message) and include location information, as defined in clause 5.6.4, a unique Call Identifier, a Message Identifier and a History-Info header value containing the original PSAP's Reply-To header value. The message type shall be `start|redirect/273`, and the message body shall contain a predefined text (e.g. introducing the person in need).

- Upon the reception of (`start|redirect/273`), the PSAP shall respond with 200 OK and issue an automatic SIP MESSAGE to return the received Call Identifier and provide a Message Identifier with a Message Type set to `start/257`. This message shall include a Reply-To header value that provides the public routable SIP URI of the PSAP and a predefined text (e.g. the first question a PSAP may ask in the chat).

- After the reception and acknowledgement of the first PSAP message (`start/257`), either end (APP or PSAP) may continue with SIP MESSAGE transactions in session mode as defined in clause 6.2.3.

Message Sequence Chart

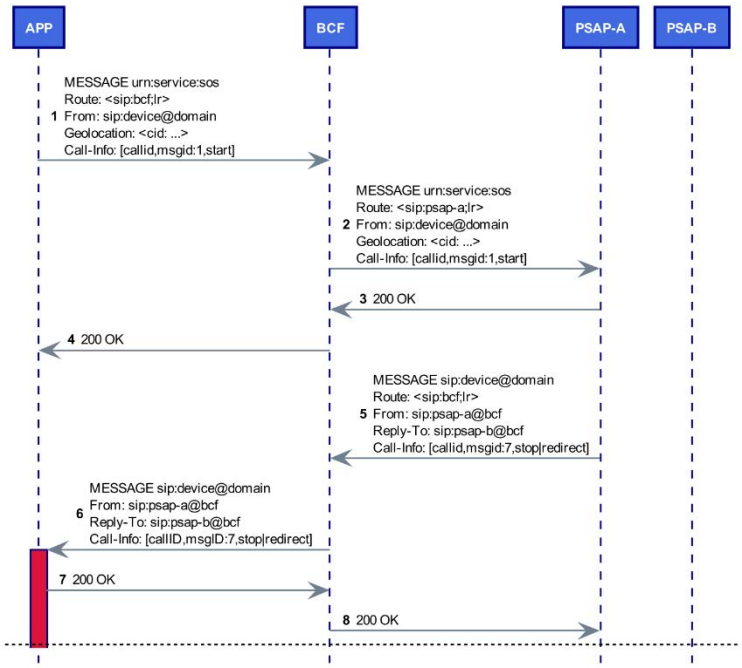

Figure 22: Redirect - Part 1

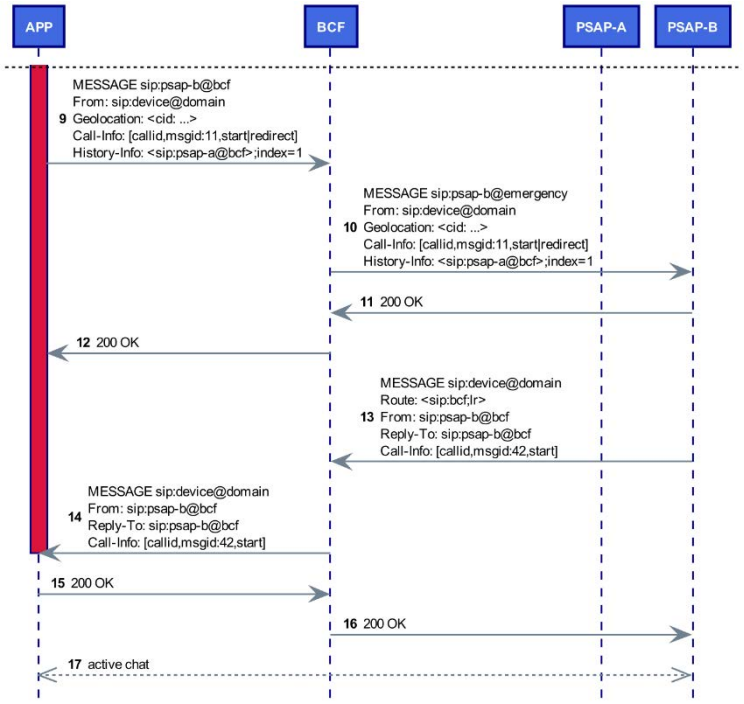

Figure 23: Redirect - Part 2

6.3 Chat Transfer (HTTP-3)

6.3.1 Overview

The Chat Transfer (HTTP-3) interface supports to transfer an active chat session from one PSAP to another. The transfer consists of two steps - first a negotiation request is sent to determine the willingness of transfer target to accept a transfer via a dedicated response. The second step is the actual execution of the transfer initiated by an execution request that includes message content (optional) and meta data of the original chat to be forwarded. The following clauses assume that PSAPs involved in a transfer have a certain agreement how a chat transfer is operationally handled.

6.3.2 Transfer Negotiation

Transfer Negotiation is a web service whereby the transfer target of an emergency chat becomes the responding entity. The requesting (chatTransferNegotiationRequest) entity provides the own SIP URI and the URI of the wanted target together with the Call Identifier of the chat being transferred and optional text. The responding entity (chatTransferNegotiationResponse) provides a Status Code and an expiration time indicating the period transfer execution requests are accepted.

Table 5: ChatTransferNegotiationRequest

Parameter	Condition	Description
CallId	MANDATORY	Call Identifier of the chat to be transferred
TransferFrom	MANDATORY	SIP URI of the transferring entity
TransferTo	MANDATORY	SIP URI of the transfer target
ReasonText	OPTIONAL	Free text to describe the transfer reason

Table 6: ChatTransferNegotiationResponse

Parameter	Condition	Description
CallId	MANDATORY	Call Identifier of the chat to be transferred
ExpiryTime	MANDATORY	Time in seconds after which the response expires
StatusCode	MANDATORY	Status Code

- Status Codes:
  - 202 Accepted
  - 406 Not Acceptable

The `ExpiryTime` in the response is the actual expiration time of the negotiation. In the case of expiry, a corresponding chatTransferExection request will be declined by the transferee with 503 Service Unavailable.

6.3.3 Transfer Execution

Transfer Execution is a web service whereby the requesting entity (chatTransferExecutionRequest) provides the Call Identifier, that last Message Identifier, an expiration time indication the period transfer execution responses are accepted, and, optionally, the complete chat history to the transfer target of an emergency chat. The responding entity (chatTransferExecutionResponse) provides a Status Code.

Table 7: ChatTransferExecutionRequest

Parameter	Condition	Description
CallId	MANDATORY	Call Identifier of the chat to be transferred
MessageId	MANDATORY	Last message ID used in this chat
ExpiryTime	MANDATORY	Time in seconds after which the request expires
Data	OPTIONAL	JSON object containing the complete chat

Table 8: ChatTransferExecutionResponse

Parameter	Condition	Description
CallId	MANDATORY	Call Identifier of the chat to be transferred
StatusCode	MANDATORY	Status Code

- Status Codes:
  - 200 OK
  - 503 Service Unavailable

The `ExpiryTime` in the request is the actual expiration time of the execution request. In the case a corresponding chatTransferExection response is still pending after expiry, the transferee shall not continue processing the transfer and the transferor shall re-activate the emergency chat.

6.3.4 Message Sequence Chart

The following message sequence charts indicate timing of asynchronous web service messages and corresponding SIP transactions. To better understand the workflow processes, actors are included, where CT-A and CT-B represent a call taker belonging to the respective PSAP. Italic text and dashed arrows indicate user interaction. Activated blocks below a participant indicate a timeline where user input is stalled (red), or expiry timers are active (salmon or pink).

Figures 24 to 26 illustrate a successful chat transfer (negotiation, execution and chat message exchange) whereas Figures 27 to 29 illustrate a transfer error caused either by a declined execution request (timeout) or other technical or operational issues. Such errors result in a reactivation of the previous chat session.

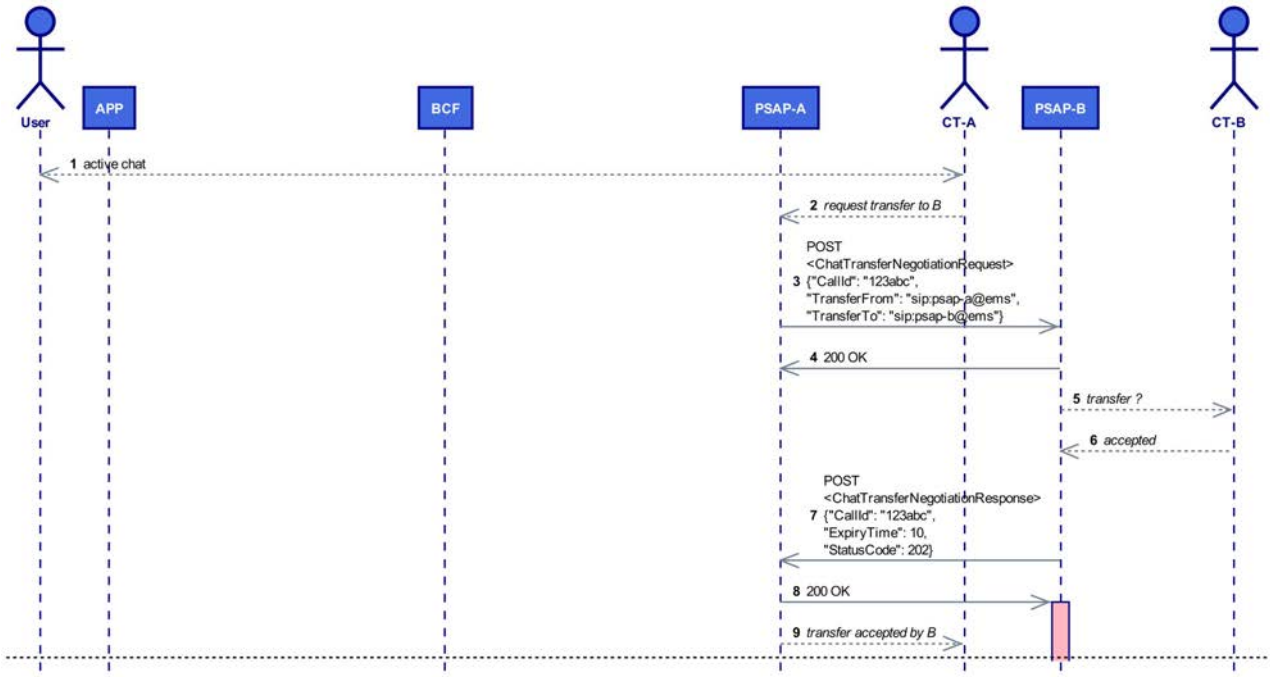

Figure 24: Chat Transfer - Part 1

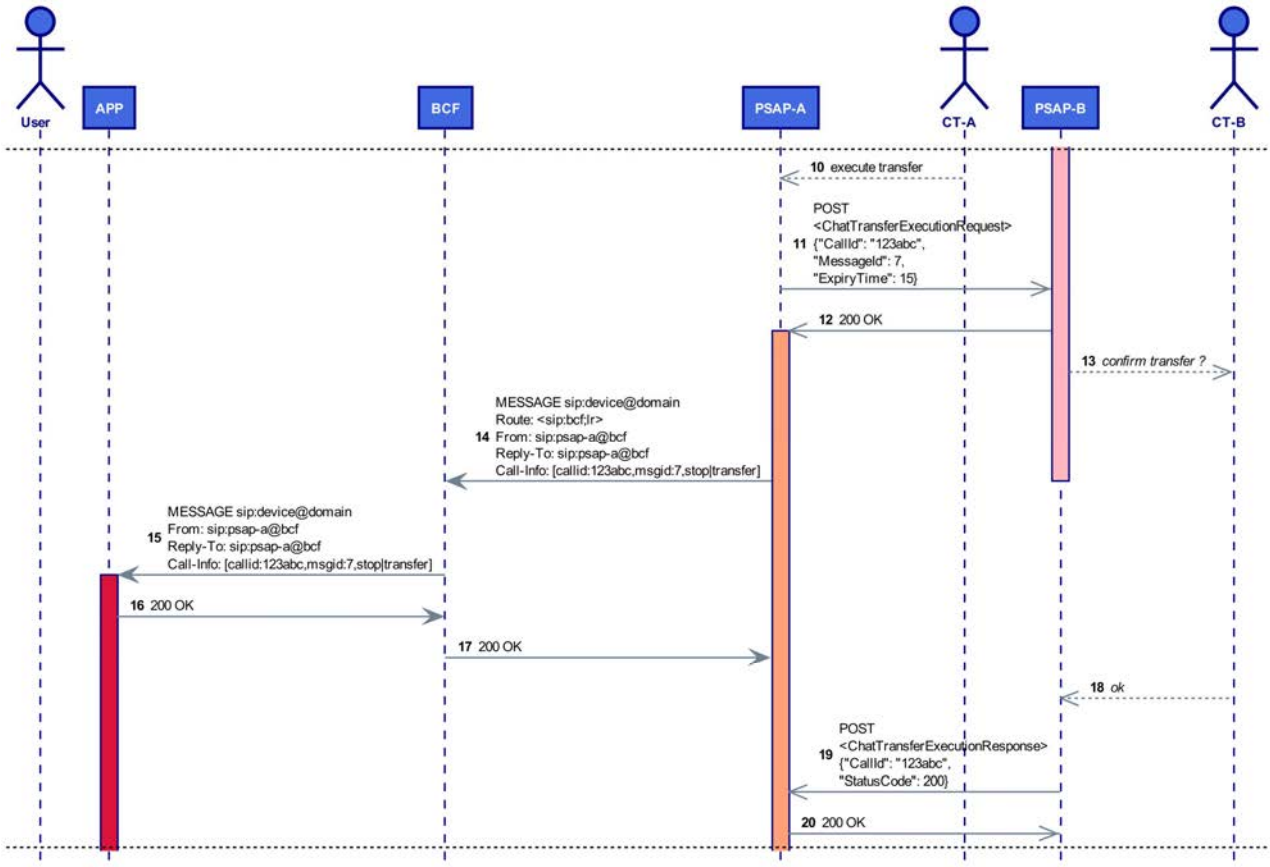

Figure 25: Chat Transfer - Part 2

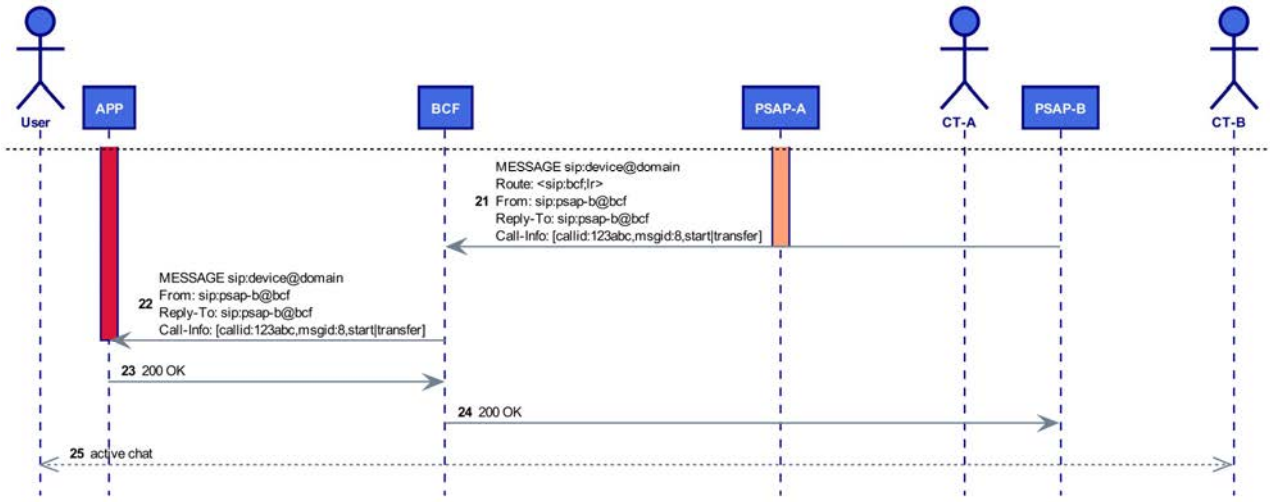

Figure 26: Chat Transfer - Part 3

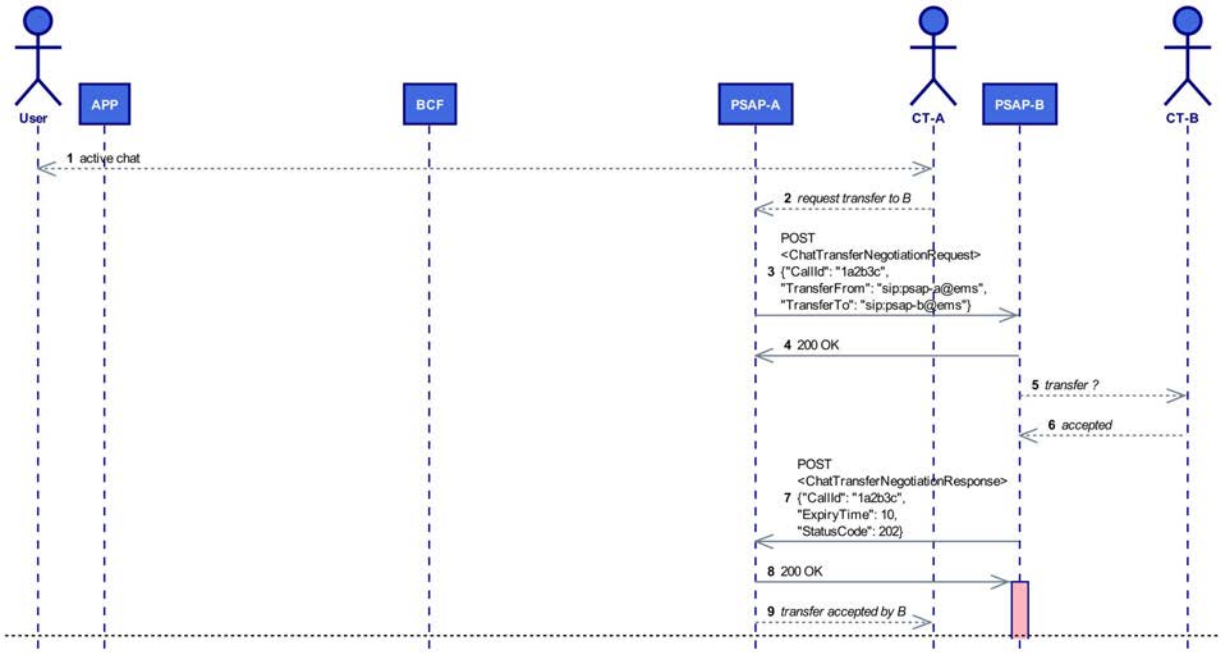

Figure 27: Chat Transfer Error - Part 1

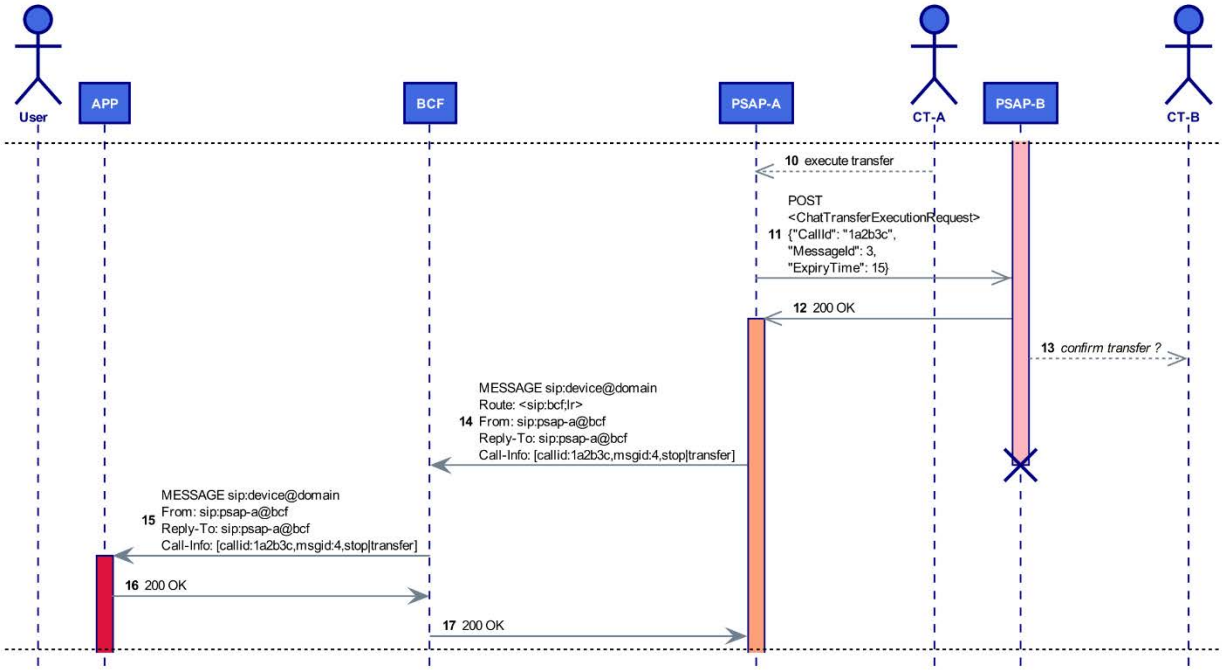

Figure 28: Chat Transfer Error - Part 2

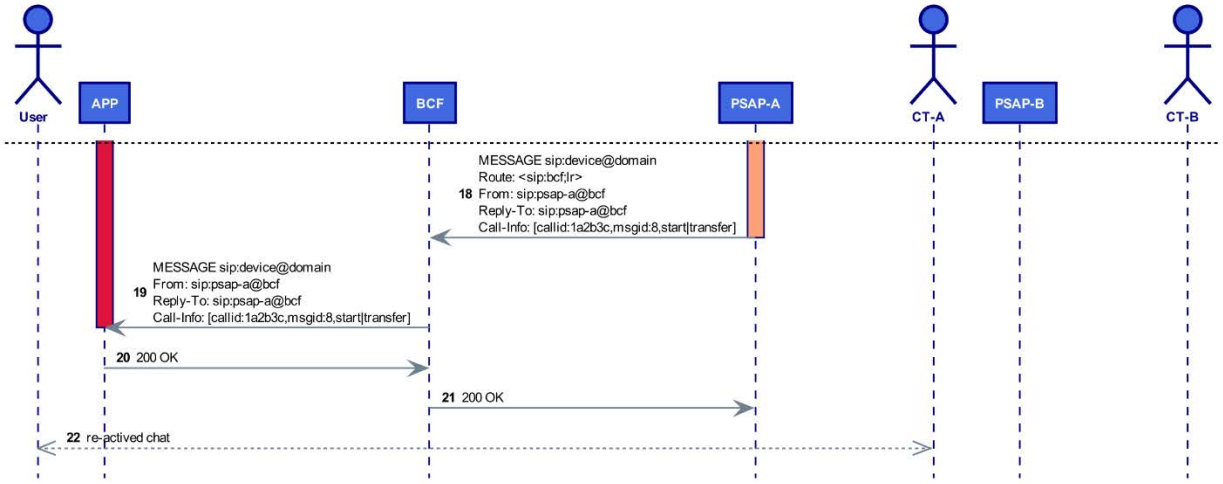

Figure 29: Chat Transfer Error - Part 3

Annex A (normative): JSON Schema

A.1 ChatTransferNegotiationRequest

```
{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://forge.etsi.org/rep/emtel/ts-103-698/json-schema/blob/v1.1.1/chattrnegrequ.json",
  "type": "object",
  "title": " ChatTransferNegotiationRequest",
  "description": "chat transfer negotiation request",
  "required": [
    "CallId",
    "TransferFrom",
    "TransferTo"
  ],
  "properties": {
    "CallId": {
      "type": "string",
      "description": "Call identifier of the chat being transferred"
    },
    "TransferFrom": {
      "type": "string",
      "description": "SIP URI of transfer initiating entity"
    },
    "TransferTo": {
      "type": "string",
      "description": "SIP URI of the transfer receiving entity"
    },
    "ReasonText": {
      "type": "string",
      "description": "Free text describing the reason of the transfer"
    }
  }
}
```

A.2 ChatTransferNegotiationResponse

```
{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://forge.etsi.org/rep/emtel/ts-103-698/json-schema/blob/v1.1.1/chattrnegresp.json",
  "type": "object",
  "title": " ChatTransferNegotiationResponse",
  "description": "chat transfer negotiation response",
  "required": [
    "CallId",
    "ExpiryTime",
    "StatusCode"
  ],
  "properties": {
    "CallId": {
      "type": "string",
      "description": "Call identifier of the chat being transferred"
    },
    "ExpiryTime": {
      "type": "integer",
      "description": " Time in seconds the negotiation response will expire",
      "minimum": 10
    },
    "StatusCode": {
      "type": "integer",
      "enum": [
        202,
        406
      ],
      "description": "Free text describing the reason of the transfer"
    }
  }
}
```

A.3 ChatTransferExecutionRequest

```
{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://forge.etsi.org/rep/emtel/ts-103-698/json-schema/blob/v1.1.1/chattrexerequ.json",
  "type": "object",
  "title": " ChatTransferExecutionRequest",
  "description": "chat transfer execution request",
  "required": [
    "CallId",
    "MessageId",
    "ExpiryTime"
  ],
  "properties": {
    "CallId": {
      "type": "string",
      "description": "Call identifier of the chat being transferred"
    },
    "MessageId": {
      "type": "integer",
      "description": "SIP URI of transfer initiating entity",
      "minimum": 1
    },
    "ExpiryTime": {
      "type": "integer",
      "description": " Time in seconds the execution request will expire",
      "minimum": 10
    },
    "Data": {
      "type": "object",
      "description": "tbd"
    }
  }
}
```

A.4 ChatTransferExecutionResponse

```
{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "https://forge.etsi.org/rep/emtel/ts-103-698/json-schema/blob/v1.1.1/chattrexeresp.json",
  "type": "object",
  "title": " ChatTransferExecutionResponse",
  "description": "chat transfer execution response",
  "required": [
    "CallId",
    "StatusCode"
  ],
  "properties": {
    "CallId": {
      "type": "string",
      "description": "Call identifier of the chat being transferred"
    },
    "StatusCode": {
      "type": "integer",
      "enum": [
        200,
        503
      ],
      "description": "Status code of the response"
    }
  }
}
```

A.5 Message Type Definition

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-----------+---+---------------+
| reserved  | v |    type       | v...version
+-----------+---+---------------+
|0 0 0 0 0 0|0 1|0 0 0 0 0 0 0 0| unknown
|0 0 0 0 0 0|0 1|0 0 0 0 0 0 0 1| start
|0 0 0 0 0 0|0 1|0 0 0 0 0 0 1 1| in-chat
|0 0 0 0 0 0|0 1|0 0 0 0 0 0 1 0| stop
|0 0 0 0 0 0|0 1|0 0 0 0 0 1 0 0| heartbeat
|0 0 0 0 0 0|0 1|0 0 0 0 1 0 x x| transfer
|0 0 0 0 0 0|0 1|0 0 0 1 0 0 x x| redirect
|0 0 0 0 0 0|0 1|0 0 1 x x x x x| reserved
|0 0 0 0 0 0|0 1|0 1 x x x x x x| reserved
|0 0 0 0 0 0|0 1|1 0 0 0 0 0 0 0| inactive
+-----------+---+---------------+
```

Source: https://forge.etsi.org/rep/emtel/ts-103-698/message-type-definition/blob/v1.1.1/lmpe-types.txt.

Annex B (informative): Organizational Descriptions

B.0 General

This clause provides a summary of the organizations described in the present document.

B.1 Certificate Authority

A Certificate Authority (CA) that issues certificates to different entities in the emergency services networks has to be created or the services of an existing CA have to be re-used. This enables proper authentication and builds the foundation for authorization. The overall level of security will be substantially improved therefore.

Since the present document assumes a public key infrastructure the use of such a certificate authority for usage with emergency services organizations is needed. Note that a CA is responsible for managing the entire lifecycle of certificates from the creation to termination or revocation.

B.2 National, and Regional Authorities

Applicable laws, regulations and rules may need to be enhanced to support ESInet deployment. This is particularly true to provide the necessary provisions to require access network providers to share IP location information and VSPs/ASPs to transmit emergency calls to emergency services authorities.

B.3 Public Safety Computer Emergency Response Team (CERT)

To react to security breaches and other incidents the creation of a Public Safety Computer Emergency Response Team (CERT) is anticipated, and all stakeholders are obliged to make any necessary preparations to receive alerts from the CERT and to respond. It is essential that all organizations have trained staff available 24 x 7 x 365 to immediately respond to attacks and have the capability and training to be able to mitigate such attacks.

B.4 ETSI Protocol Naming and Numbering Service (PNNS)

ETSI CTI provides a Protocol Naming and Numbering Service for all ETSI Technical Bodies. Many protocols require the allocation of globally unique names or numbers to interoperate successfully. Ranges of names or numbers are often allocated to standards bodies to distribute the task of allocation, while still maintaining global uniqueness. ETSI CTI manages such name and number ranges for ETSI.

B.5 Emergency Call Service Authorities

The national/regional/local authorities are responsible for overall operation of, and the data for the emergency communication system. Such an authority:

- oversees operating the state/regional/local Emergency Service Routing Proxy (ESRP);
- provides Emergency Call Routing Function (ECRF) and Location Information Service (LIS);

- is responsible for maintaining the integrity of the data housed in the ECRF systems;
- also provides input to the definition of policies, which dictates the granularity of the routing decisions returned by the ECRF (i.e. ESRP URIs vs. PSAP URIs);
- provides data about PSAP boundaries. This data is, for example, using in LoST servers and influences routing decisions;
- is responsible to address issues caused by gaps and overlaps in these boundaries;
- ensures that BCFs are accessible from the Internet so that VSPs and ASPs can route emergency calls to them;
- is responsible to provide an authoritative GIS database containing only valid in formation, where civic addresses are used for the location validation;
- decides about the setup, and operation of the ESInet as well as PSAPs and other IT infrastructure equipment necessary to operate the IP network, interconnection points, and call routing equipment.

Annex C (informative): Parameter Registries

C.0 General

The present document requires several registries to be created and those populated with initial values. The entity that creates these values and makes them available over the Web is called ETSI Protocol Naming and Numbering Service (PNNS). ETSI PNNS ensures that the policies associated with the parameter registries are followed to avoid inconsistency in the registry.

Annex D (informative): Use Case Examples

D.0 General

This clause provides LMPE use case examples.

D.1 National/Regional

The first use case represents a national or regional deployment of LMPE. According to the specification in the present document, a simplified configuration may look like illustrated in Figure D.1. Note that light grey boxes indicate functional elements outside of an ESInet.

Precondition are an ESInet and an application or RCS service provider that maintains a trunk to the local ESInet to forward SIP requests containing either a local emergency number in the request URI (e.g. *sip:112@provider.a*) or a proper service urn (e.g. *urn:service:sos*). The term trunk may not only represent a technical term for SIP peering, but also a trusted relation between the public service provider and the authority operating an ESInet.

The example includes two PSAPs serving specific regions, A1 and A2, an ECRF maintaining a mapping to internal SIP URIs and an ESRP. In addition, the BCF is configured to translate public to private domains (e.g. *public.a* ⇔ *psap.a*) when forwarding requests to a specific PSAP.

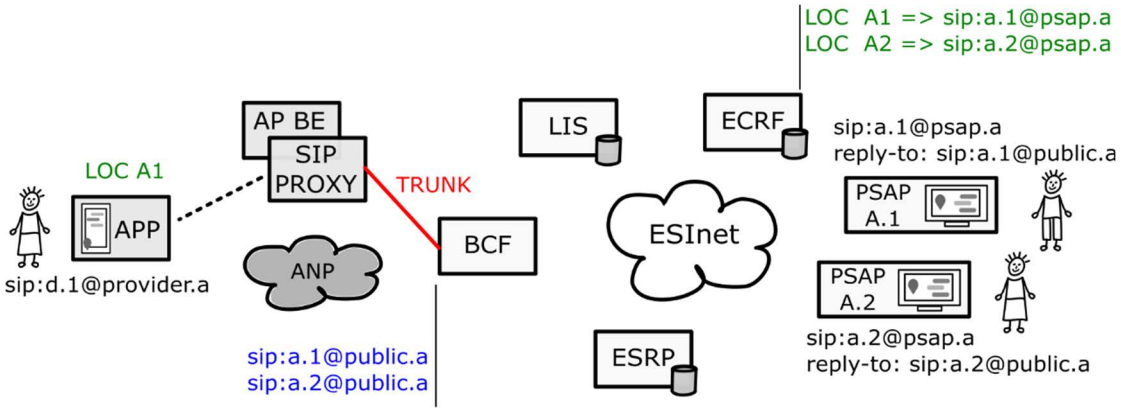

Figure D.1: National/regional LMPE deployment

In a first step, users register their device via the application backend (AP BE), most probably using an SMS verification of the mobile number to receive proper SIP settings (e.g. *sip:d.1@provider.a*) for the application. In case of emergency a user initiates an emergency chat and sends the first message (addressing a service urn) via the application provider's SIP proxy to the BCF. The BCF, by default, forwards the message to the ESRP and the ESRP queries the ECRF to get proper mapping information (sip:a.1@psap.a). In case that the originating device is located in region A1, the initial message is then forwarded by the ESRP to PSAP A.1 based on the SIP URI received via a LoST request.

The response to the first inbound message sent from the PSAP advertises its public SIP URI via the Reply-To header that points to the BCF and therefore allows direct message exchange without having the ESRP as intermediate service, even if the location of the originating device changes.

PSAPs may use the ESRP as outbound SIP proxy that has a certain policy to forward outbound messages to the proper BCF based on the requested domain (e.g. *provider.a*). Any routing operation used in the example is part of a BCF's or ESRP's default feature set.

D.2 International/Roaming

The second use case represents an international deployment of LMPE to illustrate a roaming scenario. According to the specification in the present document, a simplified configuration may look like illustrated in Figure D.2. Note that light grey boxes indicate functional elements outside of an ESInet.

Besides the given preconditions (refer to clause D.1), the following is also important. Assuming two different ESInet configurations, authorities operating an ESInet need to ensure a trustworthy peering among themselves and a proper configured Forest Guide (FG). There are several different ways to implement peering, one example how individual ESInets can enable peering with other ESInets is through BGP. In this example only logical relations that require peering are shown as orange dashed lines (as in Figure D.2).

The example includes two ESInets (A and B) each including two PSAPs serving specific regions, an ECRF maintaining a mapping to internal SIP URIs and an ESRP. In addition, the BCFs are configured to translate public to private domains (e.g. *public.a* ⇔ *psap.a* and *public.b* ⇔ *psap.b*) when forwarding requests to a specific PSAP, and to relay white listed domain names (e.g. *pubic.a*, *provider.a*, *public.b*, …).

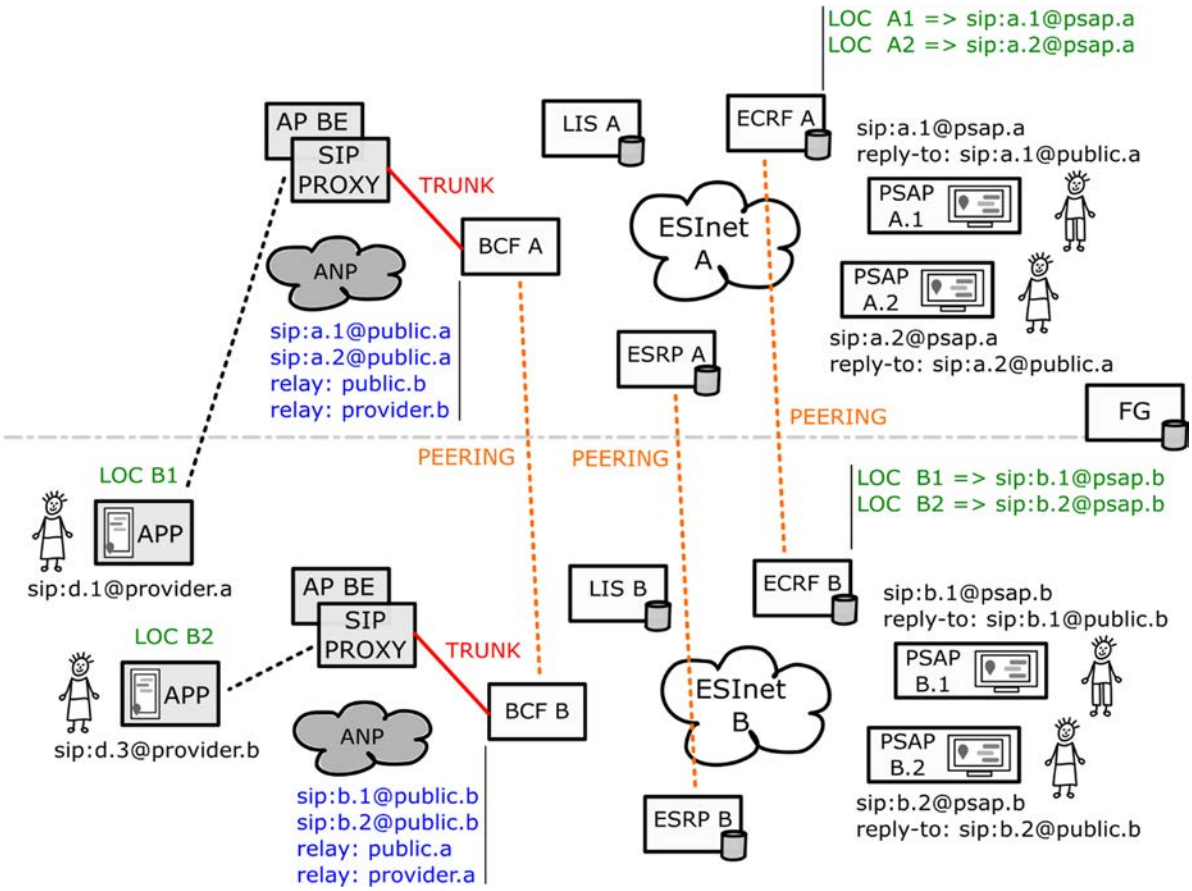

Figure D.2: International LMPE deployment (Roaming)

Again, in a first step, users register their device via the local application backend (AP BE), most probably using an SMS verification of the mobile number to receive proper SIP settings (e.g. sip:d.1@provider.a *or sip:d.3@provider.a*b) for the application. Note that users register just once and only with the local application backend, even in case of international roaming.

In case a user initiates an emergency chat within the home region, the same procedure applies as introduced in clause D.1. In case of emergency in a visiting country B (roaming) a user initiates an emergency chat and sends the first message (addressing a service urn) via the application provider's SIP proxy to the BCF of the home ESInet A. In the given example device sip:d.1@provider.a is located in country B and therefore provides a location (LOC B1) somewhere in country B. As the next step, the BCF A, by default, forwards the message to the ESRP A. which in turn queries the local ECRF A to get mapping information.

ECRF A does not have an authoritative mapping and therefore queries the FG to find out from which ECRF it may get an authoritative mapping for the given location. In this example, the FG redirects ECRF A to ECRF B to retrieve mapping information , and finally, ECRF A returns sip:b.1@psap.b as next hop SIP URI to ESRP A. The initial message is then forwarded by ESRP A to PSAP B. Note that depending on the ESInet deployment there may also be a terminating ESRP acting as next hop - to simplify the example, the message is sent directly to PSAP B.

The response to the first inbound message sent from PSAP B advertises its public SIP URI via the Reply-To header that points to BCF B and therefore allows direct message exchange without having the ESRP A and B as intermediate service, even if the location of the originating device changes. As the response contains the domain of *provider.a*, it is relayed from BCF B to BCF A based on the white listed domain names (e.g. exchanged according to a peering agreement). Any further message exchange follows the same relaying procedure at either BCF.

PSAP B may use ESRP B as outbound SIP proxy that has a certain policy to forward outbound messages to BCF B based on the requested domain (e.g. *provider.a*). Any routing operation used in the example is part of a BCF's or ESRP's default feature set.

D.3 Smart IoT Devices And Chatbots

Another use case due to the lightweight character of LMPE, is the simple integration of smart IoT devices or chatbot services as illustrated in Figure D.3. Note that light grey boxes indicate functional elements outside of an ESInet.

Especially, solutions supporting pre-emptive care (such as chatbot type services that help or try to answer questions about health care) do not require additional media channels or complex interfaces, and can nevertheless help lessen the load on human emergency services. According to [i.2], there are a lot of interesting use cases of IoT in emergency communications, and LMPE is a candidate technology for a simple and standardized transport mechanism to carry any sensors data objects to the proper PSAP.

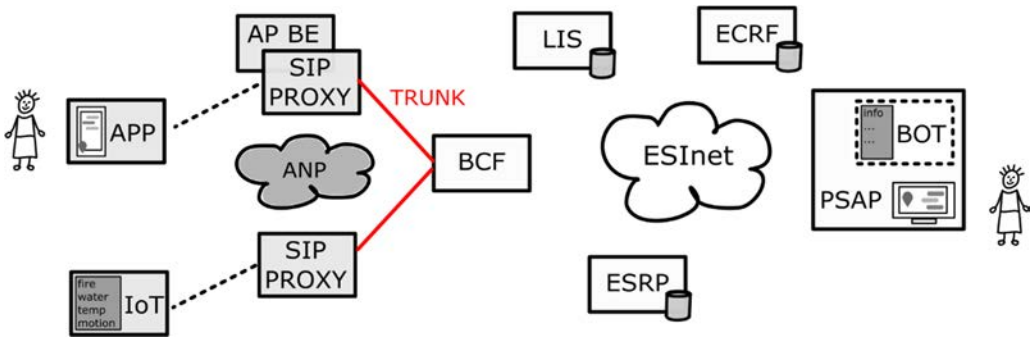

Figure D.3: Smart IoT Devices And Chatbots

History

Document history		
V1.1.1	December 2020	Publication