

Network Working Group  
INTERNET-DRAFT  
Category: Informational  
Expires: December 30, 2013

B. Aboba  
M. Thomson  
Skype  
13 June 2013

Emergency Services Support in WebRTC  
draft-aboba-rtcweb-ecrit-01.txt

## Abstract

The Web Real-Time Communication (WebRTC) framework supports interactive communication between web-browsers, including support for audio, video and text. This document describes how emergency services functionality can be implemented within the WebRTC framework, including support for location and call routing as well as interoperability with Public Safety Answering Points (PSAPs) supporting next generation emergency services.

## Legal

THIS DOCUMENT AND THE INFORMATION CONTAINED THEREIN ARE PROVIDED ON AN "AS IS" BASIS AND THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST, AND THE INTERNET ENGINEERING TASK FORCE, DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION THEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 30, 2013.

## Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
1.1 Terminology . . . . .	3
1.2 Prior Work . . . . .	4
2. Location and Call Routing Requirements . . . . .	4
3. Media Requirements . . . . .	7
4. Accessibility . . . . .	8
5. Security Considerations . . . . .	9
6. IANA Considerations . . . . .	11
7. Acknowledgments . . . . .	11
8. References . . . . .	11
8.1. Normative References . . . . .	11
8.2 Informative references . . . . .	11
Authors' Addresses . . . . .	15

## 1. Introduction

The Web Real-Time Communication (WebRTC) framework supports interactive communication between web-browsers, including support for audio, video and text. This document describes how emergency services functionality can be implemented within the WebRTC framework. Since signaling is out of scope of the WebRTC standards suite as noted in "Overview: Real Time Protocols for Browser-based Applications" [I-D.ietf-rtcweb-overview] Section 3, this document focuses on other aspects such as location, call routing and media support.

No guidance is provided as to whether a given WebRTC application or service will be subject to emergency service obligations. As noted in "Best Current Practice for Communications Services in support of Emergency Calling" [RFC6881] Section 4:

Some jurisdictions have regulations governing which devices need to support emergency calling and developers are encouraged to ensure that devices they develop meet relevant regulatory requirements. Unfortunately, the natural variation in those regulations also makes it impossible to accurately describe the cases when developers do or do not have to support emergency calling.

It should also be understood that this document does not advocate use of IP-based communication in all situations. For example, where accurate location cannot be obtained, emergency callers could be better served by utilizing the telephony capabilities of the underlying platform (e.g., a mobile-device) where available, as proposed in [WebTel]. This can enable location to be provided in situations where it would not otherwise be available, as well as permitting an emergency call to be placed even when the device does not have access to the Internet.

The document is laid out as follows: Section 1 provides an introduction and reviews prior work. Section 2 discusses requirements relating to location and call routing. Section 3 discusses media requirements. Section 4 discusses accessibility. Section 5 discusses security considerations.

### 1.1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

This document uses terms from [RFC3261], [RFC5012] and [RFC6443].

## 1.2. Prior Work

The IETF ECRIT WG has developed an overview of the emergency calling architecture as well as a best current practice document detailing implementation requirements.

"Framework for Emergency Calling using Internet Multimedia" [RFC6443] provides an overview of how IETF specifications can be used to support emergency calling using multimedia. At a high level, this involves determination of the caller location, conveyance of the location within a signaling protocol such as Session Initiation Protocol (SIP) [RFC6442], routing of the call using the Location-to-Service Translation (LoST) protocol [RFC5222], and exchange of media using Real-time Transport Protocol (RTP) [RFC3550].

"Best Current Practice for Communication Services in support of Emergency Calling" [RFC6881] builds on [RFC6443] to describe the requirements for end devices ("ED-" requirements), access networks ("AN-"), service providers ("SP-"), Public Safety Answering Points (PSAPs) and intermediate devices ("INT-") to achieve globally interoperable emergency calling on the Internet.

Both [RFC6443] and [RFC6881] assume the use of SIP as the signaling mechanism for emergency calling. As noted in [RFC6443] Section 1:

This document discusses the use of the Session Initiation Protocol (SIP) [RFC3261] by PSAPs and calling parties. While other inter-domain call signaling protocols may be used for emergency calling, SIP is ubiquitous and possesses the proper support of this use case.

Since standardization of signaling is out of scope of the WebRTC standards effort, and WebRTC applications can utilize a wide variety of signaling mechanisms, the requirements described in [RFC6881] do not necessarily apply to WebRTC implementations, applications and services. Therefore in this document, we focus on emergency calling requirements that are independent of the signaling mechanism, such as those relating to accessibility, location, call routing and media.

## 2. Location and Call Routing Requirements

Determination of caller location as well as call routing is an essential aspect of emergency services support. Relevant requirements from [RFC6881] include:

ED-15/INT-4/AN-4 Devices, intermediate Devices and/or access networks SHOULD support a manual method to override the location the access network determines. When the override location is

supplied in civic form, it MUST be possible for the resultant Presence Information Data Format - Location Object (PIDF-LO) received at the PSAP to contain any of the elements specified in [RFC4119] and [RFC5139].

ED-17/INT-9/AN-9 Devices that support endpoint measuring of location MUST have at least a coarse location capability (typically <1km accuracy) for routing of calls. The location mechanism MAY be a service provided by the access network.

ED-24 Where the operating system supporting application programs which need location for emergency calls does not allow access to Layer 2 and Layer 3 functions necessary for a client application to use DHCP location options and/or other location technologies that are specific to the type of access network, the operating system MUST provide a published API conforming to ED-12 through ED-23 and ED-25 through ED-32. It is RECOMMENDED that all operating systems provide such an API.

ED-41/SP-20 Location objects MUST be created with information about the method by which the location was determined, such as GPS, manually entered, or based on access network topology included in a PIDF- LO "method" element. In addition, the source of the location information MUST be included in a PIDF-LO "provided-by" element.

ED-49 Endpoints MUST support one or more mechanisms that allow them to determine their public IP address, for example, STUN [RFC5389].

ED-50 Endpoints MUST support LIS discovery as described in [RFC5986], and the LoST discovery as described in [RFC5223].

Since browser applications do not have direct access to operating system location APIs, ED-24 is not applicable to WebRTC.

For reasons that will be described, automatically obtaining location suitable for emergency use is challenging for WebRTC applications. In order to ensure that location is available when needed, as well as to provide resilience against errors in automated location determination, WebRTC emergency service applications SHOULD support manual override as recommended in ED-15.

The W3C Geolocation API [GeolocationAPI] was not developed with emergency services location in mind, so that requirements ED-17 and ED-41 are not well supported. [GeolocationAPI] does not provide information on the source of the location information as required in ED-41; attempting to infer the source from the accuracy parameter is

NOT RECOMMENDED. Currently, Location Based Services utilized by Geolocation APIs do not warrant their use in emergency services and do not consistently provide the accuracy required by emergency services applications, so that emergency use of the W3C Geolocation API is also NOT RECOMMENDED.

An alternative is to implement location configuration and call routing in Javascript, using an HTTP-based protocol such as HELD [RFC5985] and LoST [RFC5222]. While this approach can provide location usable in emergency services applications, it is only applicable on networks with a Location Information Server (LIS), such as enterprise deployments subject to Multi-Line Telephone System (MLTS) regulations [StateMLTS].

In order to utilize location and call routing services, it is first necessary to locate the appropriate servers. Since the discovery mechanisms described in [RFC5986] and [RFC5223] are based on use of a DHCP option, which cannot be assumed to be accessible in Javascript, ED-50 is difficult to support within WebRTC-based emergency services applications.

For LoST discovery, the emergency services application can determine the appropriate LoST server(s) on its own. To avoid potential issues, it is best to avoid pre-configuration of particular servers, allowing the appropriate server to be determined dynamically.

LIS discovery requires determination of the domain name that can be used for LIS discovery, as noted in [RFC5986] Section 3.4:

If a Device knows one or more alternative domain names that might be used for discovery, it MAY repeat the U-NAPTR process using those domain names as input. For instance, static configuration of a Device might be used to provide a Device with a domain name.

While static configuration of the domain name can be used in situations where device mobility is restricted, the appropriate LIS depends on the network to which the host is attached, so that this is not a general solution.

"Location Information Server (LIS) Discovery using IP address and Reverse DNS" [I.D.ietf-geopriv-res-gw-lis-discovery] specifies a means for a device to discover several alternative domain names that can be used as input to the Dynamic Delegation Discovery Service (DDDS). Since several of the techniques (such as use of PTR RRs and Session Traversal Utilities for NAT (STUN) [RFC5389]) are potentially implementable in WebRTC-based emergency services applications this approach MAY be used.

### 3. Media Requirements

Within [RFC6881] media-related requirements are covered in Section 14. These include:

ED-71 Endpoints MUST send and receive media streams on RTP [RFC3550].

ED-72 Normal SIP offer/answer [RFC3264] negotiations MUST be used to agree on the media streams to be used.

ED-73/SP-41 G.711 A law (and mu Law if they are intended be used in North America) encoded voice as described in [RFC3551] MUST be supported. If the endpoint cannot support G.711, a transcoder MUST be used so that the offer received at the PSAP contains G.711. It is desirable to include wideband codecs such as G.722 and AMR-WB in the offer. PSAPs SHOULD support narrowband codecs common on endpoints in their area to avoid transcoding.

ED-74 Silence suppression (Voice Activity Detection methods) MUST NOT be used on emergency calls. PSAP call takers sometimes get information on what is happening in the background to determine how to process the call.

ED-77 Endpoints supporting video MUST support H.264 per [RFC6184].

Requirement ED-71 is satisfied by compliant WebRTC implementations since "Web Real-Time Communication (WebRTC): Media Transport and Use of RTP" [I-D.ietf-rtcweb-rtp-usage] Section 4.1 requires support for RTP [RFC3550].

Requirement ED-72 is specific to SIP and so does not apply generally to WebRTC implementations, applications and services. However, it is believed that the APIs under development within the W3C WebRTC WG can support this requirement.

Requirement ED-74 is satisfied by compliant WebRTC implementations since the WebRTC APIs under development within W3C [WEBRTC], support silence suppression control via the "constraints" parameter.

[I-D.ietf-rtcweb-rtp-usage] Section 4.3 does not provide a recommendation on a mandatory-to-implement set of codecs. While ED-73 does not require implementation of G.711 if the service supports transcoding, G.711 is not difficult to implement and is widely supported, with a high level of interoperability. Therefore it is recommended that G.711 be included as a mandatory-to-implement audio codec within [I-D.ietf-rtcweb-rtp-usage] Section 4.3.

Currently the disposition of ED-77 is unclear. Discussion of mandatory-to-implement video codecs is ongoing within the IETF RTCWEB WG, but has not reached a conclusion. While there is a need to support interoperable video within emergency services applications, more options may be available within an emergency services context than would be the case for general use. For example, within the PSAP, it may be feasible to support multiple video codecs, either by installation of browser plugins, or by use of multiple browsers. In some emergency service applications (such as the VRS), codec requirements may be specific to the service and may be satisfiable by a custom device or browser approved for use with that service, which may include the required codecs implemented natively or via plug-ins, as the service provider sees fit.

#### 4. Accessibility

By lowering the barriers to development of realtime-enabled browser applications, as well as by building on accessibility support within the browser, WebRTC promises to enable the development of a new generation of accessible emergency applications and services.

In order to support accessibility, it is RECOMMENDED that WebRTC-based emergency applications and services conform to the Web Content Accessibility Guidelines (WCAG) v2.0 [WCAG].

In order to support accessibility for individuals with hearing or speech disabilities, support for textual communications is important.

Currently the W3C is developing a proposed charter for the Timed Text Working Group [TTWG], which will potentially produce a second edition of the timed Text Markup Language (TTML) 1.0 recommendation as well as publishing a recommendation for a version 1.1 specification.

Text-related requirements in [RFC6881] are covered in Section 14, including:

ED-75 Endpoints supporting Instant Messaging (IM) MUST support either [RFC3428] and [RFC4975].

ED-76 Endpoints supporting real-time text MUST use [RFC4103]. The expectations for emergency service support for the real-time text medium are described in [RFC5194], Section 7.1.

Since [RFC3428] and [RFC4975] are both based on SIP, ED-75 does not apply to all WebRTC-based emergency applications and services. As noted in "Emergency Services Functionality with the Extensible Messaging and Presence Protocol (XMPP)" [I-D.tschofenig-ecrit-xmpp-es], XMPP [RFC6120] is a potential alternative for emergency services



applications looking to support instant messaging [RFC6121] and multi-user chat [XEP-045] functionality.

"RTP Payload for Text Conversation" [RFC4103] is typically implemented along with SIP signaling as described in "Framework for Real-Time Text over IP Using the Session Initiation Protocol (SIP)" [RFC5194]. As a result, ED-76 does not apply to WebRTC implementations.

Alternatives to support of real-time text functionality are available, such as "In-Band Real Time Text" [XEP-0301], which supports real-time text by addition of child elements within XMPP message stanzas. The use of child elements to encapsulate real-time text, as well as transmission of complete lines enables [XEP-0301] to provide backward compatibility with existing XMPP instant-messaging and Multi-User Chat (MUC) clients, with no changes required to XMPP servers. Since XMPP can be encapsulated within HTTP via mechanisms such as BOSH [XEP-0206] or WebSockets [RFC6455], [XEP-0301] can be implemented in Javascript. Experience with Javascript implementation using the [Strophe] XMPP library indicates that adequate performance is achievable. In contrast, implementing real-time text as media as in [RFC4103] requires native browser support, as well as requiring changes to the configuration of intermediaries such as Session Border Controllers (SBCs). Also, [RFC4103] is not backward compatible with SIP instant messaging implementations supporting page-mode [RFC3428] or session [RFC4975] approaches.

## 5. Security Considerations

Security requirements in [RFC6881] include:

ED-48/SP-24 TLS [RFC5746] MUST be used to protect location (but see Section 9.1). All implementations MUST support TLS.

ED-58/SP-30 TLS is the primary mechanism used to secure the signaling for emergency calls. IPsec [RFC4301] MAY be used instead of TLS for any hop. Either TLS or IPSEC MUST be used when attempting to signal an emergency call.

ED-59/SP-31 If TLS session establishment is not available, or fails, the call MUST be retried without TLS.

ED-60/SP-32 [RFC5626] is RECOMMENDED to maintain persistent TLS connections between entities when one of the entity is an endpoint. Persistent TLS connection between proxies is RECOMMENDED using any suitable mechanism.

ED-61/AN-28 TLS SHOULD be used when attempting to retrieve

location (configuration or dereferencing) with HELD. The use of [RFC5077] is RECOMMENDED to minimize the time to establish TLS sessions without keeping server-side state. IPsec MAY be used instead of TLS.

ED-62/AN-29 When TLS session establishment fails, the location retrieval MUST be retried without TLS.

For WebRTC, HTTPS MUST be used to protect signaling for an emergency call, with potential fail-over to HTTP. HTTPS SHOULD be used to protect location retrieval (HELD) and call routing (LoST).

WebRTC security considerations are discussed in "Security Considerations for RTC-Web" [I-D.ietf-rtcweb-security]. The WebRTC security architecture, described in "RTCWEB Security Architecture" [I-D.ietf-rtcweb-security-arch], requires implementation of Secure RTP [RFC3711] as well as DTLS/SRTP [RFC5764].

While the security features of WebRTC exceed the requirements outlined in [RFC6881], support for emergency services within WebRTC raises concerns about potential attacks on the emergency services infrastructure, given the potential for malicious code to be executed within the browser. One way to lessen the likelihood of attacks by untrusted Javascript applications is for PSAPs to put up their own sites for emergency calling, protected by HTTPS.

While ICE [RFC5245] provides demonstration of liveness and consent to receive, it is possible for an attacker to overwhelm the PSAP by generating a large number of prank calls. IP relay services are also potential targets since these don't require forging of Caller-Id nor do they provide audio or video from the attacker.

Security threats to IP-based emergency services are described in "Security Threats and Requirements for Emergency Call Marking and Mapping" [RFC5069]. These include attacks on the emergency services system, such as attempting to deny system services to all users in a given area, to gain fraudulent use of services and to divert emergency calls to non-emergency sites. [RFC5069] also describes attacks against individuals, including attempts to prevent an individual from receiving aid, or to gain information about an emergency.

"Threat Analysis of the Geopriv Protocol" [RFC3694] describes threats against geographic location privacy, including protocol threats, threats resulting from the storage of geographic location data, and threats posed by the abuse of information.

Overall, experience indicates a relationship between anonymity and

the prevalence of prank calling. Therefore some protection may be provided through authentication of the caller either in the signaling or media plane. It is NOT RECOMMENDED that WebRTC-based emergency applications and services support anonymous emergency calling.

## 6. IANA Considerations

This document does not require actions by IANA.

## 7. Acknowledgments

We would like to thank the members of the IETF RTCWEB Working Group for discussions related to this topic.

## 8. References

### 8.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC6881] Rosen, B. and J. Polk, "Best Current Practice for Communications Services in Support of Emergency Calling", RFC 6881, March 2013.
- [WCAG] Caldwell, B., Cooper, M., Reid, L.G. and G. Vanderheiden, "Web Content Accessibility Guidelines (WCAG) 2.0", <http://www.w3.org/TR/WCAG20/>, December 2008.

### 8.2. Informative References

- [GeolocationAPI] Popescu, A., "Geolocation API Specification", W3C, <http://dev.w3.org/geo/api/spec-source.html>
- [I.D.ietf-geopriv-res-gw-lis-discovery] Thomson, M. and R. Bellis, "Location Information Server (LIS) Discovery using IP address and Reverse DNS", draft-ietf-geopriv-res-gw-lis-discovery-05 (work in progress), April 2013.
- [I-D.ietf-rtcweb-overview] Alvestrand, H., "Overview: Real Time Protocols for Browser-based Applications", draft-ietf-rtcweb-overview-06 (work in progress), February 2013.
- [I-D.ietf-rtcweb-rtp-usage] Perkins, C., Westerlund, M. and J. Ott, "Web Real-Time

Communication (WebRTC): Media Transport and Use of RTP", draft-ietf-rtcweb-rtp-usage-06 (work in progress), February 2013.

[I-D.ietf-rtcweb-security]

Rescorla, E., "Security Considerations for RTC-Web", draft-ietf-rtcweb-security-04 (work in progress), January 2013.

[I-D.ietf-rtcweb-security-arch]

Rescorla, E., "RTCWEB Security Architecture", draft-ietf-rtcweb-security-arch-06 (work in progress), July 2013.

[I-D.tschofenig-ecrit-xmpp-es]

Tschofenig, H., "Emergency Services Functionality with the Extensible Messaging and Presence Protocol (XMPP)", draft-tschofenig-ecrit-xmpp-es-00 (work in progress), March 2012.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[RFC3264] Rosenberg, J. and R. Schulzrinne, "An Offer/Answer Model with the Session Description Protocol (SDP)", RFC 3264, June 2002.

[RFC3428] Campbell, B., Rosenberg, J., Schulzrinne, H., Huitema, C. and D. Gurle, "Session Initiation Protocol (SIP) Extension for Instant Messaging", RFC 3428, December 2002.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

[RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, July 2003.

[RFC3694] Danley, M., Mulligan, D., Morris, J. and J. Peterson, "Threat Analysis of the Geopriv Protocol", RFC 3694, February 2004.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E. and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[RFC4103] Hellstrom, G. and P. Jones, "RTP Payload for Text Conversation", RFC 4103, June 2005.

- [RFC4119] Peterson, J., "A Presence-based GEOPRIV Location Object Format", RFC 4119, December 2005.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.
- [RFC4975] Campbell, B., Mahy, R. and C. Jennings, "The Message Session Relay Protocol (MSRP)", RFC 4975, September 2007.
- [RFC5012] Schulzrinne, H. and R. Marshall, "Requirements for Emergency Context Resolution with Internet Technologies", RFC 5012, January 2008.
- [RFC5069] Taylor, T., Tschofenig, H., Schulzrinne, H. and M. Shanmugam, "Security Trheats and Requirements for Emergency Call Marking and Mapping", RFC 5069, January 2008.
- [RFC5077] Salowey, J., Zhou, H., Eronen, P. and H. Tschofenig, "Transport Layer Security (TLS) Session Resumption without Server-Side State", RFC 5077, January 2008.
- [RFC5139] Thomson, M. and J. Winterbottom, "Revised Civic Location Format for Presence Information Data Format Location Object (PIDF-LO)", RFC 5139, February 2008.
- [RFC5194] van Wijk, A. and G. Gybels, "Framework for Real-Time Text over IP Using the Session Initiation Protocol (SIP)", RFC 5194, June 2008.
- [RFC5222] Hardie, T., Newton, A., Schulzrinne, H. and H. Tschofenig, "LoST: A Location-to-Service Translation Protocol", RFC 5222, August 2008.
- [RFC5223] Schulzrinne, H., Polk, J. and H. Tschofenig, "Discovering Location-to-Service Translation (LoST) Servers Using the Dynamic Host Configuration Protocol (DHCP)", RFC 5223, August 2008.
- [RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, April 2010.
- [RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT", RFC 5389, October 2008.
- [RFC5626] Jennings, C., Mahy, R. and F. Audet, "Managing Client-Initiated Connections in the Session Initiation Protocol (SIP)", RFC 5626, October 2009.

- [RFC5746] Rescorla, E., Ray, M., Dispensa, S. and N. Oskov, "Transport Layer Security (TLS) Renegotiation Indication Extension", RFC 5746, February 2010.
- [RFC5764] McGrew, D. and E. Rescorla, "Datagram Transport Layer Security (DTLS) Extension to Establish Keys for the Secure Real-time Transport Protocol (SRTP)", RFC 5764, May 2010.
- [RFC5985] Barnes, M., "HTTP Enabled Location Delivery (HELD)", RFC 5985, September 2010.
- [RFC5986] Thomson, M. and J. Winterbottom, "Discovering the Local Location Information Server (LIS)", RFC 5986, September 2010.
- [RFC6120] Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Core", RFC 6120, March 2011.
- [RFC6121] Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence", RFC 6121, March 2011.
- [RFC6184] Wang, Y.-K., Even, R., Kristensen, T. and R. Jesup, "RTP Payload Format for H.264 Video", RFC 6184, May 2011.
- [RFC6442] Polk, J., Rosen, B. and J. Peterson, "Location Conveyance for the Session Initiation Protocol", RFC 6442, December 2011.
- [RFC6443] Rosen, B., Schulzrinne, H., Polk, J. and A. Newton, "Framework for Emergency Calling Using Internet Multimedia", RFC 6443, December 2011.
- [RFC6455] Fette, I. and A. Melnikov, "The WebSocket Protocol", RFC 6455, December 2011.
- [StateMLTS] "State E911 Legislation",  
<http://www1.911enable.com/resource-center/state-e911-legislation>
- [Strophe] "Libraries for XMPP Poets", <http://strophe.im>
- [TTWG] "Proposed Timed Text Working Group Charter",  
<http://www.w3.org/2012/02/timed-text-wg-charter.html>
- [WEBRTC] Bergkvist, A., Burnett, D., Jennings, C. and A. Narayanan, "WebRTC 1.0: Real-time Communication Between Browsers", W3C Editor's Draft (work in progress),

<http://dev.w3.org/2011/webrtc/editor/webrtc.html>, June 2013.

- [WebTel] WebAPI/WebTelephony,  
<https://wiki.mozilla.org/WebAPI/WebTelephony>
- [XEP-0206] Paterson, I. and P. Saint-Andre, "XMPP Over BOSH", XEP-0206 version 1.3, <http://xmpp.org/extensions/xep-0206.html>, July 2010.
- [XEP-0301] Rejhon, M., "In-Band Real Time Text", XEP-0301 version 0.2, <http://xmpp.org/extensions/xep-0301.html>, March 2012.
- [XEP-045] Saint-Andre, P., "Multi-User Chat", XEP 0045 version 1.25, <http://xmpp.org/extensions/xep-0045.html>, February 2012.

#### Authors' Addresses

Bernard Aboba  
Skype  
Redmond, WA 98052  
US

E-Mail: [bernard\\_aboba@hotmail.com](mailto:bernard_aboba@hotmail.com)

Martin Thomson  
Skype  
3210 Porter Drive  
Palo Alto, CA 94304  
US

Phone: +1 650-353-1925  
Email: [martin.thomson@gmail.com](mailto:martin.thomson@gmail.com)

ECRIT  
Internet-Draft  
Updates: 6443, 6881 (if approved)  
Intended status: Standards Track  
Expires: October 7, 2016

R. Gellens  
  
B. Rosen  
NeuStar  
H. Tschofenig

R. Marshall  
TeleCommunication Systems, Inc.  
J. Winterbottom  
April 5, 2016

Additional Data Related to an Emergency Call  
draft-ietf-ecrit-additional-data-38.txt

Abstract

When an emergency call is sent to a Public Safety Answering Point (PSAP), the originating device, the access network provider to which the device is connected, and all service providers in the path of the call have information about the call, the caller or the location which is helpful for the PSAP to have in handling the emergency. This document describes data structures and mechanisms to convey such data to the PSAP. The intent is that every emergency call carry as much as possible of the information described here using the mechanisms described here.

The mechanisms permit the data to be conveyed by reference (as an external resource) or by value (within the body of a SIP message or a location object). This follows the tradition of prior emergency services standardization work where data can be conveyed by value within the call signaling (i.e., in the body of the SIP message) or by reference.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."



This Internet-Draft will expire on October 7, 2016.

#### Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

#### Table of Contents

1. Introduction . . . . .	4
2. Terminology . . . . .	6
3. Document Scope . . . . .	7
4. Data Structures . . . . .	7
4.1. Data Provider Information . . . . .	9
4.1.1. Data Provider String . . . . .	9
4.1.2. Data Provider ID . . . . .	10
4.1.3. Data Provider ID Series . . . . .	10
4.1.4. Type of Data Provider . . . . .	11
4.1.5. Data Provider Contact URI . . . . .	12
4.1.6. Data Provider Languages(s) Supported . . . . .	13
4.1.7. xCard of Data Provider . . . . .	14
4.1.8. Subcontractor Principal . . . . .	14
4.1.9. Subcontractor Priority . . . . .	15
4.1.10. ProviderInfo Example . . . . .	15
4.2. Service Information . . . . .	17
4.2.1. Service Environment . . . . .	18
4.2.2. Service Type . . . . .	19
4.2.3. Service Mobility Environment . . . . .	20
4.2.4. EmergencyCallData.ServiceInfo Example . . . . .	21
4.3. Device Information . . . . .	22
4.3.1. Device Classification . . . . .	22
4.3.2. Device Manufacturer . . . . .	23
4.3.3. Device Model Number . . . . .	24
4.3.4. Unique Device Identifier . . . . .	24
4.3.5. Device/Service-Specific Additional Data Structure . . . . .	25
4.3.6. Device/Service-Specific Additional Data Structure Type . . . . .	26
4.3.7. EmergencyCallData.DeviceInfo Example . . . . .	26

4.4.	Owner/Subscriber Information . . . . .	27
4.4.1.	Subscriber Data Privacy Indicator . . . . .	27
4.4.2.	xCard for Subscriber's Data . . . . .	28
4.4.3.	EmergencyCallData.SubscriberInfo Example . . . . .	28
4.5.	Comment . . . . .	31
4.5.1.	Comment . . . . .	31
4.5.2.	EmergencyCallData.Comment Example . . . . .	31
5.	Issues with getting new types of data into use . . . . .	32
5.1.	Choosing between defining a new type of block or new type of device/service-specific additional data . . . . .	32
6.	Data Transport Mechanisms . . . . .	33
6.1.	Transmitting Blocks using Call-Info . . . . .	35
6.2.	Transmitting Blocks by Reference using the <provided-by> Element . . . . .	37
6.3.	Transmitting Blocks by Value using the <provided-by> Element . . . . .	38
6.4.	The Content-Disposition Parameter . . . . .	39
7.	Examples . . . . .	41
8.	XML Schemas . . . . .	53
8.1.	EmergencyCallData.ProviderInfo XML Schema . . . . .	53
8.2.	EmergencyCallData.ServiceInfo XML Schema . . . . .	55
8.3.	EmergencyCallData.DeviceInfo XML Schema . . . . .	56
8.4.	EmergencyCallData.SubscriberInfo XML Schema . . . . .	58
8.5.	EmergencyCallData.Comment XML Schema . . . . .	59
8.6.	provided-by XML Schema . . . . .	60
9.	Security Considerations . . . . .	62
10.	Privacy Considerations . . . . .	64
11.	IANA Considerations . . . . .	67
11.1.	Emergency Call Additional Data Registry . . . . .	67
11.1.1.	Provider ID Series Registry . . . . .	67
11.1.2.	Service Environment Registry . . . . .	68
11.1.3.	Service Type Registry . . . . .	68
11.1.4.	Service Mobility Registry . . . . .	69
11.1.5.	Type of Provider Registry . . . . .	69
11.1.6.	Device Classification Registry . . . . .	69
11.1.7.	Device ID Type Registry . . . . .	70
11.1.8.	Device/Service Data Type Registry . . . . .	70
11.1.9.	Emergency Call Data Types Registry . . . . .	70
11.2.	'EmergencyCallData' Purpose Parameter Value . . . . .	72
11.3.	URN Sub-Namespace Registration for <provided-by> Registry Entry . . . . .	72
11.4.	MIME Registrations . . . . .	72
11.4.1.	MIME Content-type Registration for 'application/EmergencyCallData.ProviderInfo+xml' . . .	72
11.4.2.	MIME Content-type Registration for 'application/EmergencyCallData.ServiceInfo+xml' . . .	73
11.4.3.	MIME Content-type Registration for 'application/EmergencyCallData.DeviceInfo+xml' . . .	75

11.4.4.	MIME Content-type Registration for 'application/EmergencyCallData.SubscriberInfo+xml'	76
11.4.5.	MIME Content-type Registration for 'application/EmergencyCallData.Comment+xml'	77
11.5.	URN Sub-Namespace Registration	78
11.5.1.	Registration for urn:ietf:params:xml:ns:EmergencyCallData	78
11.5.2.	Registration for urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo	79
11.5.3.	Registration for urn:ietf:params:xml:ns:EmergencyCallData:ServiceInfo	79
11.5.4.	Registration for urn:ietf:params:xml:ns:EmergencyCallData:DeviceInfo	80
11.5.5.	Registration for urn:ietf:params:xml:ns:EmergencyCallData:SubscriberInfo	81
11.5.6.	Registration for urn:ietf:params:xml:ns:EmergencyCallData:Comment	82
11.6.	Schema Registrations	83
11.7.	VCARD Parameter Value Registration	84
12.	Acknowledgments	85
13.	References	85
13.1.	Normative References	85
13.2.	Informational References	87
13.3.	URIs	89
Appendix A.	XML Schema for vCard/xCard	90
Appendix B.	XML Validation	112
Authors' Addresses		112

## 1. Introduction

When an IP-based emergency call is initiated, a rich set of data from multiple data sources is conveyed to the Public Safety Answering Point (PSAP). This data includes information about the calling party identity, the multimedia capabilities of the device, the request for emergency services, location information, and meta-data about the sources of the data. In addition, the device, the access network provider, and any service provider in the call path has even more information that is useful for a PSAP when handling an emergency.

This document extends the basic set of data communicated with a Session Initiation Protocol (SIP) based emergency call, as described in [RFC6443] and [RFC6881], in order to carry additional data which is useful to an entity or call taker handling the call. This data is "additional" to the basic information found in the emergency call signaling used. The intent is that every emergency call carry as

much as possible of the information described here using the mechanisms described here.

This document defines three categories of this additional data that can be transmitted with an emergency call:

**Data Associated with a Location:** Primary location data is conveyed in the Presence Information Data Format Location Object (PIDF-LO) data structure as defined in RFC 4119 [RFC4119] and extended by RFC 5139 [RFC5139] and RFC 6848 [RFC6848] (for civic location information), RFC 5491 [RFC5491] and RFC 5962 [RFC5962] (for geodetic location information), and [RFC7035] (for relative location). This primary location data identifies the location or estimated location of the caller. However, there might exist additional, secondary data which is specific to the location, such as floor plans, tenant and building owner contact data, heating, ventilation and air conditioning (HVAC) status, etc. Such secondary location data is not included in the location data structure but can be transmitted using the mechanisms defined in this document. Although this document does not define any structures for such data, future documents can do so following the procedures defined here.

**Data Associated with a Call:** While some information is carried in the call setup procedure itself (as part of the SIP headers as well as in the body of the SIP message), there is additional data known by the device making the call, the access network to which the device is connected, and service providers along the path of the call. This information includes service provider contact information, subscriber identity and contact information, the type of service the service provider and the access network provide, what type of device is being used, etc. Some data is broadly applicable, while other data is dependent on the type of device or service. For example, a medical monitoring device might have sensor data. The data structures defined in this document (Data Provider Information, Device Information, and Owner/Subscriber Information) all fall into the category of "Data Associated with a Call". Note that the Owner/Subscriber Information includes the subscriber's vCard, which might contain personal information such as birthday, anniversary, etc., but the data block itself is still considered to be about the call, not the caller.

**Data Associated with a Caller:** This is personal data about a caller, such as medical information and emergency contact data. Although this document does not define any structures within this category, future documents can do so following the procedures defined here.

While this document defines data structures only within the category of Data Associated with a Call, by establishing the overall framework of Additional Data, along with general mechanisms for transport of such data, extension points and procedures for future extensions, it minimizes the work needed to carry data in the other categories. Other specifications can make use of the facilities provided here.

For interoperability, there needs to be a common way for the information conveyed to a PSAP to be encoded and identified. Identification allows emergency services authorities to know during call processing which types of data are present and to determine if they wish to access it. A common encoding allows the data to be successfully accessed.

This document defines an extensible set of data structures, and mechanisms to transmit this data either by value or by reference, either in the Session Initiation Protocol (SIP) call signaling or in the Presence Information Data Format Location Object (PIDF-LO). The data structures are usable by other communication systems and transports as well. The data structures are defined in Section 4, and the transport mechanisms (using SIP and HTTPS) are defined in Section 6.

Each data structure described in this document is encoded as a "block" of information. Each block is an XML structure with an associated Multipurpose Internet Mail Extensions (MIME) media type for identification within transport such as SIP and HTTPS. The set of blocks is extensible. Registries are defined to identify the block types that can be used and to allow blocks to be included in emergency call signaling.

Much of the information supplied by service providers and devices is private and confidential; service providers and devices generally go to lengths to protect this information; disclosing it in the context of an emergency call is a trade-off to protect the greater interest of the customer in an emergency.

## 2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

This document also uses terminology from [RFC5012]. We use the term service provider to refer to an Application Service Provider (ASP). A Voice Service Provider (VSP) is a special type of ASP. With the term "Access Network Provider" we refer to the Internet Access Provider (IAP) and the Internet Service Provider (ISP) without

further distinguishing these two entities, since the difference between the two is not relevant for this document. Note that the roles of ASP and access network provider might be provided by a single company. An Emergency Services Provider is an entity directly involved in providing emergency services. This includes PSAPs, dispatch, police, fire, emergency medical, other responders, and other similar agencies.

Within each data block definition (see Section 4), the values for the "Use:" label are specified as one of the following:

'Required': means it MUST be present in the data structure.

'Conditional': means it MUST be present if the specified condition(s) is met. It MAY be present if the condition(s) is not met.

'Optional': means it MAY be present.

vCard [RFC6350] is a data format for representing and exchanging a variety of information about individuals and other entities. For applications that use XML, the format defined in vCard is not immediately applicable. For this reason, an XML-based encoding of the information elements defined in the vCard specification has been defined and the name of that specification is xCard [RFC6351]. Since the term vCard is more familiar to most readers, we use the terms xCard and vCard interchangeably.

### 3. Document Scope

The scope of this document is explicitly limited to emergency calls. The data structures defined here are not appropriate to be conveyed in non-emergency calls because they carry sensitive and private data. However, in certain private-use situations between a specialized service provider (such as a vehicle telematics service provider) and dedicated equipment (such as in a vehicle) where the endpoints have a preexisting relationship and privacy issues are addressed within the relationship, the mechanisms and data structures defined here can be used with communications within the limited context of the preexisting relationship.

### 4. Data Structures

This section defines the following five data structures, each as a data block. For each block we define the MIME media type, and the XML encoding. The five data structures are:

'Data Provider': This block supplies name and contact information for the entity that created the data. Section 4.1 provides the details.

'Service Information': This block supplies information about the service. The description can be found in Section 4.2.

'Device Information': This block supplies information about the device placing the call. Device information can be found in Section 4.3.

'Owner/Subscriber': This block supplies information about the owner of the device or about the subscriber. Details can be found in Section 4.4.

'Comment': This block provides a way to supply free form human readable text to the PSAP or emergency responders. This simple structure is defined in Section 4.5.

Each block contains a mandatory <DataProviderReference> element. The purpose of the <DataProviderReference> element is to associate all blocks added by the same data provider as a unit. The <DataProviderReference> element associates the data provider block to each of the other blocks added as a unit. Consequently, when a data provider adds additional data to an emergency call (such as device information) it MUST add information about itself (via the data provider block) and the blocks added contain the same value in the <DataProviderReference> element. All blocks added by a single entity at the same time MUST have the same <DataProviderReference> value. (In certain situations, the same provider might process a call more than once, likely in different roles, and in such cases, each time it processes the call, it adds a new set of blocks with a new <DataProviderReference> value.) The value of the <DataProviderReference> element has the same syntax and properties (specifically, world-uniqueness) as the value of the "Message-ID" message body header field specified in RFC 5322 [RFC5322] except that the <DataProviderReference> element is not enclosed in brackets (the "<" and ">" symbols are omitted). In other words, the value of a <DataProviderReference> element is syntactically a msg-id as specified in RFC 5322 [RFC5322].

Each block is added to the Additional Data Blocks Registry created in Section 11.1.9 and categorized as providing data about the caller. New blocks added to the registry in the future MUST also be categorized per the description of the three categories in Section 1. See Section 5 and Section 5.1 for additional considerations when adding new blocks or types of data.

Note that the xCard format is re-used in some of the data structures to provide contact information. In an xCard there is no way to specify a "main" telephone number (that is, a primary or main contact number, typically of an enterprise, as opposed to a direct dial number of an individual). These numbers are useful to emergency responders who are called to a large enterprise. This document adds a new parameter value called 'main-number' to the "TYPE" parameter of the "tel" property. It can be used in any xCard in an emergency call additional data block.

#### 4.1. Data Provider Information

This block is intended to be supplied by any service provider in the path of the call, or the access network provider, and the device. It includes identification and contact information. This block MUST be supplied by any entity that provides any other block; it SHOULD be supplied by every service provider in the call path and by the access network provider if those entities do not add any other blocks. Devices SHOULD use this block to provide identifying information. The MIME media type is "application/EmergencyCallData.ProviderInfo+xml". An access network provider SHOULD provide this block either by value or by reference in the <provided-by> element of a PIDF-LO

##### 4.1.1. Data Provider String

Data Element: Data Provider String

Use: Conditional. Optional for blocks supplied by the originating device, mandatory otherwise.

XML Element: <DataProviderString>

Description: This is a plain text string suitable for displaying the name of the service provider that supplied the data structure. If the device creates the structure, it SHOULD use the value of the contact header field in the SIP INVITE.

Reason for Need: Inform the call taker of the identity of the entity providing the data.

How Used by Call Taker: Allows the call taker to interpret the data in this structure. The source of the information often influences how the information is used, believed or verified.



## 4.1.1.2. Data Provider ID

Data Element: Data Provider ID

Use: Conditional. Optional for blocks supplied by the originating device, mandatory otherwise. This data MUST be provided by all entities other than the originating device in order to uniquely identify the service provider or access provider.

XML Element: <ProviderID>

Description: A jurisdiction-specific code for, or the fully-qualified domain name of, the access network provider or service provider shown in the <DataProvidedBy> element that created the structure. NOTE: The value SHOULD be assigned by an organization appropriate for the jurisdiction. In the U.S., if the provider is registered with NENA, the provider's NENA Company ID MUST appear here. Additional information can be found at NENA Company Identifier Program [1] or NENA Company ID [2]. The NENA Company ID MUST be in the form of a URI in the following format: urn:nena:companyid:<NENA Company ID>. If the organization does not have an identifier registered with a jurisdiction-specific emergency services registrar (such as NENA), then the value MAY be the fully-qualified domain name of the service provider or access provider. The device MAY use its IP address or fully-qualified domain name (and set the "Data Provider ID Series" element to "domain").

Reason for Need: Inform the call taker of the identity of the entity providing the data.

How Used by Call Taker: Where jurisdictions have lists of providers the Data Provider ID provides useful information about the data source. The Data Provider ID uniquely identifies the source of the data, which might be needed especially during unusual circumstances and for routine logging.

## 4.1.1.3. Data Provider ID Series

Data Element: Data Provider ID Series

Use: Conditional. Optional for blocks supplied by the originating device, mandatory otherwise.

XML Element: <ProviderIDSeries>

Description: Identifies the issuer of the <ProviderID>. The Provider ID Series Registry created in Section 11.1.1 initially contains the entries shown in Figure 1.

Reason for Need: Identifies how to interpret the Data Provider ID. The combination of ProviderIDSeries and ProviderID MUST be globally unique.

How Used by Call Taker: Determines which provider ID registry to consult for more information

Name	Source	URL
NENA	National Emergency Number Association	<a href="http://www.nena.org">http://www.nena.org</a>
EENA	European Emergency Number Association	<a href="http://www.eena.org">http://www.eena.org</a>
domain	(The ID is a fully-qualified domain name)	(not applicable)

Figure 1: Provider ID Series Registry

#### 4.1.4. Type of Data Provider

Data Element: Type of Data Provider

Use: Required.

XML Element: <TypeOfProvider>

Description: Identifies the type of data provider supplying the data. The registry containing all valid values is created in Section 11.1.5 and the initial set of values is shown in Figure 2.

Reason for Need: Identifies the category of data provider.

How Used by Call Taker: This information can be helpful when deciding whom to contact when further information is needed.

Token	Description
Client	Originating client/device
Access Network Provider	Access network service provider
Telecom Provider	Telecom service provider (including native and over-the-top VoIP services)
Telematics Provider	A sensor-based service provider, especially vehicle-based
Language Translation Provider	A spoken language translation service
Emergency Service Provider	An emergency service provider conveying information to another emergency service provider.
Emergency Modality Translation	An emergency-call-specific modality translation service e.g., for sign language
Relay Provider	An interpretation service, e.g., video relay for sign language interpretation
Other	Any other type of service provider

Figure 2: Type of Data Provider Registry

## 4.1.5. Data Provider Contact URI

Data Element: Data Provider Contact URI

Use: Required

XML Element: <ContactURI>

Description: When provided by a service provider or an access network provider, this information is expected to be a URI to a 24/7 support organization tasked to provide PSAP support for this emergency call. When provided by a device, this MUST be the contact information of the user or owner of the device. (Ideally, this is the contact information of the device user, but when the owner and user are separate (e.g., the device owner is an organization), this MAY be the contact information of the owner.) The Data Provider Contact URI SHOULD be a TEL URI [RFC3966] in E.164 format fully specified with country code. If a TEL URI is not available, a generic SIP URI is acceptable. Note that this contact information is not used by PSAPs for callbacks (a call from a PSAP directly related to a recently terminated emergency

call, placed by the PSAP using a SIP Priority header field set to "psap-callback", as described in [RFC7090]).

Reason for Need: Additional data providers might need to be contacted in error cases or other unusual circumstances.

How Used by Call Taker: To contact the supplier of the additional data for assistance in handling the call.

#### 4.1.6. Data Provider Language(s) Supported

Data Element: Data Provider Language(s) supported

Use: Required.

XML Element: <Language>

Description: This field encodes the language used by the entity at the Data Provider Contact URI. The content of this field consists of a single token from the language tags registry, which can be found at [LanguageTagRegistry], and is defined in [RFC5646]. Multiple instances of this element MAY occur but the order is significant and the preferred language SHOULD appear first. The content MUST reflect the languages supported at the contact URI.

(Note that this field informs the PSAP of the language(s) used by the data provider. If the PSAP needs to contact the data provider, it can be helpful to know in advance the language(s) used by the data provider. If the PSAP uses a communication protocol to reach the data provider, that protocol might have language facilities of its own (such as the 'language' media feature tag, defined in RFC 3840 [RFC3840] and the more extensive language negotiation mechanism proposed with [I-D.ietf-slim-negotiating-human-language]), and if so, those are independent of this field.)

Reason for Need: This information indicates if the emergency service authority can directly communicate with the service provider or if an interpreter will be needed.

How Used by Call Taker: If the call taker cannot speak any language supported by the service provider, a translation service will need to be added to the conversation. Alternatively, other persons at the PSAP, besides the call taker, might be consulted for help (depending on the urgency and the type of interaction).

#### 4.1.1.7. xCard of Data Provider

Data Element: xCard of Data Provider

Use: Optional

XML Element: <DataProviderContact>

Description: Per [RFC6351] the xCard structure is represented within a <vcard> element. Although multiple <vcard> elements can be contained in a structure only one <vcard> element SHOULD be provided. If more than one appears, the first SHOULD be used. There are many fields in the xCard and the creator of the data structure is encouraged to provide all available information. N, ORG, ADR, TEL, EMAIL are suggested at a minimum. N SHOULD contain the name of the support group or device owner as appropriate. If more than one TEL property is provided, a parameter from the vCard Property Value registry SHOULD be specified for each TEL. For encoding of the vCard this specification uses the XML-based encoding specified in [RFC6351], referred to in this document as "xCard".

Reason for Need: Information needed to determine additional contact information.

How Used by Call Taker: Assists the call taker by providing additional contact information aside from what is included in the SIP INVITE or the PIDF-LO.

#### 4.1.1.8. Subcontractor Principal

When the entity providing the data is a subcontractor, the Data Provider Type is set to that of the primary service provider and this entry is supplied to provide information regarding the subcontracting entity.

Data Element: Subcontractor Principal

Use: Conditional. This data is required if the entity providing the data is a subcontractor.

XML Element: <SubcontractorPrincipal>

Description: Some providers outsource their obligations to handle aspects of emergency services to specialized providers. If the data provider is a subcontractor to another provider this element contains the DataProviderString of the service provider to indicate which provider the subcontractor is working for.

Reason for Need: Identify the entity the subcontractor works for.

How Used by Call Taker: Allows the call taker to understand what the relationship between data providers and the service providers in the path of the call are.

#### 4.1.1.9. Subcontractor Priority

Data Element: Subcontractor Priority

Use: Conditional. This data is required if the entity providing the data is a subcontractor.

XML Element: <SubcontractorPriority>

Description: If the subcontractor is supposed to be contacted first then this element MUST have the value "sub". If the provider the subcontractor is working for is supposed to be contacted first then this element MUST have the value "main".

Reason for Need: Inform the call taker whom to contact first, if support is needed.

How Used by Call Taker: To decide which entity to contact first if assistance is needed.

#### 4.1.1.10. ProviderInfo Example

```
<?xml version="1.0" encoding="UTF-8"?>
<ad:EmergencyCallData.ProviderInfo
  xmlns:ad="urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo">
  <ad:DataProviderReference>string0987654321@example.org
  </ad:DataProviderReference>
  <ad:DataProviderString>Example VoIP Provider
  </ad:DataProviderString>
  <ad:ProviderID>urn:nena:companyid:ID123</ad:ProviderID>
  <ad:ProviderIDSeries>NENA</ad:ProviderIDSeries>
  <ad:TypeOfProvider>Telecom Provider</ad:TypeOfProvider>
  <ad:ContactURI>tel:+1-201-555-0123</ad:ContactURI>
  <ad:Language>en</ad:Language>
  <ad:DataProviderContact
    xmlns="urn:ietf:params:xml:ns:vcard-4.0">
    <vcard>
      <fn><text>Hannes Tschofenig</text></fn>
      <n>
        <surname>Hannes</surname>
        <given>Tschofenig</given>
```

```
<additional/>
<prefix/>
<suffix>Dipl. Ing.</suffix>
</n>
<bday><date>--0203</date></bday>
<anniversary>
  <date-time>20090808T1430-0500</date-time>
</anniversary>
<gender><sex>M</sex></gender>
<lang>
  <parameters><pref><integer>1</integer></pref>
  </parameters>
  <language-tag>de</language-tag>
</lang>
<lang>
  <parameters><pref><integer>2</integer></pref>
  </parameters>
  <language-tag>en</language-tag>
</lang>
<org>
  <parameters><type><text>work</text></type>
  </parameters>
  <text>Example VoIP Provider</text>
</org>
<adr>
  <parameters>
    <type><text>work</text></type>
    <label><text>Hannes Tschofenig
      Linnoitustie 6
      Espoo , Finland
      02600</text></label>
  </parameters>
  <pobox/>
  <ext/>
  <street>Linnoitustie 6</street>
  <locality>Espoo</locality>
  <region>Uusimaa</region>
  <code>02600</code>
  <country>Finland</country>
</adr>
<tel>
  <parameters>
    <type>
      <text>work</text>
      <text>voice</text>
    </type>
  </parameters>
  <uri>tel:+358 50 4871445</uri>
```

```

</tel>
<tel>
  <parameters>
    <type>
      <text>work</text>
      <text>main-number</text>
      <text>voice</text>
    </type>
  </parameters>
  <uri>tel:+358 50 5050505</uri>
</tel>
<email>
  <parameters><type><text>work</text></type>
</parameters>
  <text>hannes.tschofenig@nsn.com</text>
</email>
<geo>
  <parameters><type><text>work</text></type>
</parameters>
  <uri>geo:60.210796,24.812924</uri>
</geo>
<key>
  <parameters><type><text>home</text></type>
</parameters>
  <uri>
    http://www.tschofenig.priv.at/key.asc
  </uri>
</key>
<tz><text>Finland/Helsinki</text></tz>
<url>
  <parameters><type><text>home</text></type>
</parameters>
  <uri>http://www.tschofenig.priv.at</uri>
</url>
</vcard>
</ad:DataProviderContact>
</ad:EmergencyCallData.ProviderInfo>

```

Figure 3: EmergencyCallData.ProviderInfo Example.

#### 4.2. Service Information

This block describes the service that the service provider provides to the caller. It SHOULD be included by all service providers in the path of the call. The MIME media type is "application/EmergencyCallData.ServiceInfo+xml".



#### 4.2.1. Service Environment

Data Element: Service Environment

Use: Conditional: Required unless the 'ServiceType' value is 'wireless'.

XML Element: <ServiceEnvironment>

Description: This element indicates whether a call is from a business or residence. Currently, the only valid entries are 'Business', 'Residence', and 'unknown', as shown in Figure 4. New values can be defined via the registry created in Section 11.1.2.

Reason for Need: To provide context and a hint when determining equipment and manpower requirements.

How Used by Call Taker: Information can be used to provide context and a hint to assist in determining equipment and manpower requirements for emergency responders. This is non-authoritative: There are situations where the service provider does not know the type of service (e.g., anonymous pre-paid service). The type of service does not necessarily reflect the nature of the premises (e.g., a business line installed in a residence, or cellular service). The registry does not contain all possible values for all situations. Hence, this is at best advisory information, but since it mimics a similar capability in some current emergency calling systems (e.g., a field in the Automatic Location Information (ALI) information used with legacy North American wireline systems), it is known to be valuable to PSAPs. The service provider uses its best information (such as a rate plan, facilities used to deliver service or service description) to determine the information and is not responsible for determining the actual characteristics of the location from which the call originated. Because the usefulness is unknown (and less clear) for cellular, this element is OPTIONAL for commercial mobile radio services (e.g., cellular) and REQUIRED otherwise.

Token	Description
Business	Business service
Residence	Residential service
unknown	Type of service unknown (e.g., anonymous pre-paid service)

Figure 4: Service Environment Registry

#### 4.2.2. Service Type

Data Element: Service Delivered by Provider to End User

Use: Required

XML Element: <ServiceType>

Description: This defines the type of service over which the call is placed (similar to the Class of Service delivered with legacy emergency calls in some some regions). The implied mobility of this service cannot be relied upon. A registry is created in Section 11.1.3. The initial set of values is shown in Figure 5. More than one value MAY be returned. For example, a VoIP inmate telephone service is a reasonable combination.

Reason for Need: Knowing the type of service can assist the PSAP in handling of the call.

How Used by Call Taker: Call takers often use this information to determine what kinds of questions to ask callers, and how much to rely on supportive information. As the information is not always available, and the registry is not all-encompassing, this is at best advisory information, but since it mimics a similar capability in some legacy emergency calling systems, it is known to be valuable.

Name	Description
wireless	Wireless Telephone Service: Includes CDMA, GSM, Wi-Fi, WiMAX, LTE (but not satellite)
coin	Fixed public pay/coin telephones: Any coin or credit card operated device
one-way	One way outbound service
temp	Soft dial tone/quick service/warm disconnect/suspended
MLTS-hosted	Hosted multi-line telephone system such as Centrex
MLTS-local	Local multi-line telephone system, includes all PBX, key systems, Shared Tenant Service
sensor-unattended	These are devices that generate DATA ONLY. This is a one-way information transmit without interactive media
sensor-attended	Devices that are supported by a monitoring service provider or that are capable of supporting interactive media
POTS	Wireline: Plain Old Telephone Service
OTT	An over-the-top service that provides communication over arbitrary Internet access (fixed, nomadic, mobile)
digital	Wireline non-OTT digital phone service
OPX	Off-premise extension
relay	A service where a human third-party agent provides additional assistance. This includes sign language relay/interpretation, telematics services that provide a human on the call, and similar services

Figure 5: Service Delivered by Provider to End User Registry

The initial set of values has been collected from sources of currently-used systems, including [NENA-02-010], [nc911], [NANP], and [LERG].

#### 4.2.3. Service Mobility Environment

Data Element: Service Mobility Environment

Use: Required

XML Element: <ServiceMobility>

Description: This provides the service provider's view of the mobility of the caller's device. As the service provider might not know the characteristics of the actual device or access network used, the value should be treated as advisory and not be relied upon. A registry is created in Section 11.1.4 with the initial valid entries shown in Figure 6.

Reason for Need: Knowing the service provider's belief of mobility can assist the PSAP with the handling of the call.

How Used by Call Taker: To determine whether to assume the location of the caller might change.

Token	Description
Mobile	The device is able to move at any time
Fixed	The device is not expected to move unless the service is relocated
Nomadic	The device is not expected to change its point of attachment while on a call
Unknown	No information is known about the service mobility environment for the device

Figure 6: Service Mobility Registry

#### 4.2.4. EmergencyCallData.ServiceInfo Example

```
<?xml version="1.0" encoding="UTF-8"?>
<svc:EmergencyCallData.ServiceInfo
  xmlns:svc="urn:ietf:params:xml:ns:EmergencyCallData:ServiceInfo">
  <svc:DataProviderReference>2468.IBOC.MLTS.1359@example.org
  </svc:DataProviderReference>
  <svc:ServiceEnvironment>Business</svc:ServiceEnvironment>
  <svc:ServiceType>MLTS-hosted</svc:ServiceType>
  <svc:ServiceMobility>Fixed</svc:ServiceMobility>
</svc:EmergencyCallData.ServiceInfo>
```

Figure 7: EmergencyCallData.ServiceInfo Example.

#### 4.3. Device Information

This block provides information about the device used to place the call. It SHOULD be provided by any service provider that knows what device is being used, and by the device itself. The MIME media type is "application/EmergencyCallData.DeviceInfo+xml".

##### 4.3.1. Device Classification

Data Element: Device Classification

Use: Optional

XML Element: <DeviceClassification>

Description: This data element defines the kind of device making the emergency call. If the device provides the data structure, the device information SHOULD be provided. If the service provider provides the structure and it knows what the device is, the service provider SHOULD provide the device information. Often the carrier does not know what the device is. It is possible to receive two Device Information blocks, one provided by the device and one from the service provider. This information describes the device, not how it is being used. This data element defines the kind of device making the emergency call. A registry is created in Section 11.1.6 with the initial set of values as shown in Figure 8.

Reason for Need: The device classification implies the capability of the calling device and assists in identifying the meaning of the emergency call location information that is being presented. For example, does the device require human intervention to initiate a call or is this call the result of programmed instructions? Does the calling device have the ability to update location or condition changes? Is this device interactive or a one-way reporting device?

How Used by Call Taker: Can provide the call taker context regarding the caller, the capabilities of the calling device or the environment in which the device is being used, and can assist in understanding the location information and capabilities of the calling device. For example, a cordless handset might be outside or next door.

Token	Description
cordless	Cordless handset
fixed	Fixed phone
satellite	Satellite phone
sensor-fixed	Fixed (non mobile) sensor/alarm device
desktop	Soft client on desktop PC
laptop	Soft client on laptop type device
tablet	Soft client on tablet type device
alarm-monitored	Alarm system
sensor-mobile	Mobile sensor device
aircraft	Aircraft telematics device
automobile	Automobile/cycle/off-road telematics
truck	Truck/construction telematics
farm	Farm equipment telematics
marine	Marine telematics
personal	Personal telematics device
feature-phone	Feature- (not smart-) cellular phone
smart-phone	Smart-phone cellular phone (native)
smart-phone-app	Soft client app on smart-phone
unknown-device	Soft client on unknown device type
game	Gaming console
text-only	Other text device
NA	Not Available

Figure 8: Device Classification Registry Initial Values

#### 4.3.2. Device Manufacturer

Data Element: Device Manufacturer

Use: Optional

XML Element: <DeviceMfgr>

Description: The plain language name of the manufacturer of the device.

Reason for Need: Used by PSAP management for post-mortem investigation/resolution.

How Used by Call Taker: Probably not used by the calltaker, but by PSAP management.

#### 4.3.3. Device Model Number

Data Element: Device Model Number

Use: Optional

XML Element: <DeviceModelNr>

Description: Model number of the device.

Reason for Need: Used by PSAP management for after-action investigation/resolution.

How Used by Call Taker: Probably not used by the calltaker, but by PSAP management.

#### 4.3.4. Unique Device Identifier

Data Element: Unique Device Identifier

Use: Optional

XML Element: <UniqueDeviceID>

XML Attribute: <TypeOfDeviceID>

Description: A string that identifies the specific device (or the device's current SIM) making the call or creating an event. Note that more than one <UniqueDeviceID> can be present, to supply more than one of the identifying values.

The <TypeOfDeviceID> attribute identifies the type of device identifier. A registry is created in Section 11.1.7 with an initial set of values shown in Figure 9.

Reason for Need: Uniquely identifies the device (or, in the case of IMSI, a SIM), independent of any signaling identifiers present in the call signaling stream.

How Used by Call Taker: Probably not used by the call taker; might be used by PSAP management during an investigation. (For example, if a PSAP experiences repeated false/accidental calls and there is

no callback number or it isn't usable, the PSAP might need to try and track down the device using various means (e.g., contacting service providers in the area). In the case of handsets without current service, it might be possible to determine who last had service. Another example might be a disconnected call where the call taker believes there is a need for assistance but was not able to obtain a location or other information).

Example: `<UniqueDeviceID TypeOfDeviceID="SN">12345</UniqueDeviceID>`

Token	Description
MEID	Mobile Equipment Identifier (CDMA)
ESN	Electronic Serial Number (GSM)
MAC	Media Access Control Address (IEEE)
WiMAX	Device Certificate Unique ID
IMEI	International Mobile Equipment ID (GSM)
IMSI	International Mobile Subscriber ID (GSM)
UDI	Unique Device Identifier
RFID	Radio Frequency Identification
SN	Manufacturer Serial Number

Figure 9: Registry of Device Identifier Types

#### 4.3.5. Device/Service-Specific Additional Data Structure

Data Element: Device/service-specific additional data structure

Use: Optional

XML Element: `<DeviceSpecificData>`

Description: A URI representing additional data whose schema is specific to the device or service which created it. (For example, a medical device or medical device monitoring service might have a defined set of medical data). The URI, when dereferenced, MUST yield a data structure defined by the Device/service-specific additional data type value. Different data can be created by each classification; e.g., a medical device created data set.

Reason for Need: Provides device/service-specific data that can be used by the call taker and/or responders.

How Used by Call Taker: Provide information to guide call takers to select appropriate responders, give appropriate pre-arrival instructions to callers, and advise responders of what to be



prepared for. May be used by responders to guide assistance provided.

#### 4.3.6. Device/Service-Specific Additional Data Structure Type

Data Element: Type of device/service-specific additional data structure

Use: Conditional: MUST be provided when a device/service-specific additional URI is provided

XML Element: <DeviceSpecificType>

Description: A value from the registry defined in Section 11.1.8 to describe the type of data located at the device/service-specific additional data structure. The initial values shown in Figure 10 currently only include IEEE 1512, which is the USDOT model for traffic incidents.

Reason for Need: This data element allows identification of externally defined schemas, which might have additional data that can assist in emergency response.

How Used by Call Taker: This data element allows the end user (call taker or first responder) to know what type of additional data is available to aid in providing the needed emergency services.

Note: This mechanism is not appropriate for information specific to a location or a caller (person).

Token	Description	Specification
IEEE1512	Common Incident Management Message Set (USDOT model for traffic incidents)	IEEE 1512-2006

Figure 10: Device/Service Data Type Registry

The IEEE 1512-2006 specifications can be found at [IEEE-1512-2006].

#### 4.3.7. EmergencyCallData.DeviceInfo Example

```
<?xml version="1.0" encoding="UTF-8"?>
<dev:EmergencyCallData.DeviceInfo
  xmlns:dev="urn:ietf:params:xml:ns:EmergencyCallData:DeviceInfo">
  <dev:DataProviderReference>d4b3072df.201409182208075@example.org
  </dev:DataProviderReference>
  <dev:DeviceClassification>fixed</dev:DeviceClassification>
  <dev:DeviceMfgr>Nokia</dev:DeviceMfgr>
  <dev:DeviceModelNr>Lumia 800</dev:DeviceModelNr>
  <dev:UniqueDeviceID TypeOfDeviceID="IMEI">35788104
  </dev:UniqueDeviceID>
</dev:EmergencyCallData.DeviceInfo>
```

Figure 11: EmergencyCallData.DeviceInfo Example.

#### 4.4. Owner/Subscriber Information

This block describes the owner of the device (if provided by the device) or the subscriber information (if provided by a service provider). The contact location is not necessarily the location of the caller or incident, but is rather the nominal contact address. The MIME media type is "application/EmergencyCallData.SubscriberInfo+xml".

In some jurisdictions some or all parts of the subscriber-specific information are subject to privacy constraints. These constraints vary but dictate which information can be displayed and logged. A general privacy indicator expressing a desire for privacy by the subscriber is provided. The interpretation of how this is applied is left to the receiving jurisdiction as the custodians of the local regulatory requirements. This matches an equivalent privacy flag provided in some legacy emergency call systems.

##### 4.4.1. Subscriber Data Privacy Indicator

Attribute: 'privacyRequested', Boolean.

Use: Conditional. This attribute MUST be provided if the owner/subscriber information block is not empty.

Description: The subscriber data privacy indicator specifically expresses the subscriber's desire for privacy. In some jurisdictions subscriber services can have a specific "Type of Service" which prohibits information, such as the name of the subscriber, from being displayed. This attribute is provided to explicitly indicate whether the subscriber service includes such constraints. The interpretation of this indicator is left to each jurisdiction (in keeping with the semantics of the privacy indicator provided in some legacy emergency call systems).

Because the interpretation of this indicator varies based on local regulations, this document cannot describe the exact semantics nor indicate which fields are affected (the application of this indicator might affect the display of data contained in any of the blocks).

Reason for Need: Some jurisdictions require subscriber privacy to be observed when processing emergency calls.

How Used by Call Taker: Where privacy is indicated the call taker might not have access to some aspects of the subscriber information.

#### 4.4.2. xCard for Subscriber's Data

Data Element: xCARD for Subscriber's Data

Use: Conditional. Subscriber data MUST be provided unless it is not available. Some services, such as prepaid phones, non-initialized phones, etc., do not have information about the subscriber.

XML Element: <SubscriberData>

Description: Information known by the service provider or device about the subscriber; e.g., Name, Address, Individual Telephone Number, Main Telephone Number and any other data. <n>, <org> (if appropriate), <adr>, <tel>, <email> are suggested at a minimum. If more than one <tel> property is provided, a parameter from the vCard Property Value registry MUST be specified on each <tel>. While some data (such as <anniversary>) might not seem obviously relevant for emergency services, any data is potentially useful in some emergency circumstances.

Reason for Need: When the caller is unable to provide information, this data can be used to obtain it

How Used by Call Taker: Obtaining critical information about the caller and possibly the location when it is not able to be obtained otherwise. While the location here is not necessarily that of caller, in some circumstances it can be helpful in locating the caller when other means have failed.

#### 4.4.3. EmergencyCallData.SubscriberInfo Example

```
<?xml version="1.0" encoding="UTF-8"?>
<sub:EmergencyCallData.SubscriberInfo
  xmlns:sub=
```

```
"urn:ietf:params:xml:ns:EmergencyCallData:SubscriberInfo"
  privacyRequested="false">
<sub:DataProviderReference>FEABFECD901@example.org
</sub:DataProviderReference>
<sub:SubscriberData xmlns="urn:ietf:params:xml:ns:vcard-4.0">
  <vcard>
    <fn><text>Simon Perreault</text></fn>
    <n>
      <surname>Perreault</surname>
      <given>Simon</given>
      <additional/>
      <prefix/>
      <suffix>ing. jr</suffix>
      <suffix>M.Sc.</suffix>
    </n>
    <bday><date>--0203</date></bday>
    <anniversary>
      <date-time>20090808T1430-0500</date-time>
    </anniversary>
    <gender><sex>M</sex></gender>
    <lang>
      <parameters><pref><integer>1</integer></pref>
      </parameters>
      <language-tag>fr</language-tag>
    </lang>
    <lang>
      <parameters><pref><integer>2</integer></pref>
      </parameters>
      <language-tag>en</language-tag>
    </lang>
    <org>
      <parameters><type><text>work</text></type>
      </parameters>
      <text>Viagenie</text>
    </org>
    <adr>
      <parameters>
        <type><text>work</text></type>
        <label><text>Simon Perreault
          2875 boul. Laurier, suite D2-630
          Quebec, QC, Canada
          G1V 2M2</text></label>
      </parameters>
      <pobox/>
      <ext/>
      <street>2875 boul. Laurier,
        suite D2-630</street>
      <locality>Quebec</locality>
```

```
<region>QC</region>
<code>G1V 2M2</code>
<country>Canada</country>
</adr>
<tel>
  <parameters>
    <type>
      <text>work</text>
      <text>voice</text>
    </type>
  </parameters>
  <uri>tel:+1-418-656-9254;ext=102</uri>
</tel>
<tel>
  <parameters>
    <type>
      <text>work</text>
      <text>voice</text>
      <text>main-number</text>
    </type>
  </parameters>
  <uri>tel:+1-418-555-0000</uri>
</tel>
<tel>
  <parameters>
    <type>
      <text>work</text>
      <text>text</text>
      <text>voice</text>
      <text>cell</text>
      <text>video</text>
    </type>
  </parameters>
  <uri>tel:+1-418-262-6501</uri>
</tel>
<email>
  <parameters><type><text>work</text></type>
</parameters>
  <text>simon.perreault@viagenie.ca</text>
</email>
<geo>
  <parameters><type><text>work</text></type>
</parameters>
  <uri>geo:46.766336,-71.28955</uri>
</geo>
<key>
  <parameters><type><text>work</text></type>
</parameters>
```

```
        <uri>
          http://
            www.viagenie.ca/simon.perreault/simon.asc
        </uri>
      </key>
      <tz><text>America/Montreal</text></tz>
      <url>
        <parameters><type><text>home</text></type>
        </parameters>
        <uri>http://nomis80.org</uri>
      </url>
    </vcard>
  </sub:SubscriberData>
</sub:EmergencyCallData.SubscriberInfo>
```

Figure 12: EmergencyCallData.SubscriberInfo Example.

#### 4.5. Comment

This block provides a mechanism for the dataprovider to supply extra, human readable information to the PSAP. It is not intended for a general purpose extension mechanism nor does it aim to provide machine-readable content. The MIME media type is "application/EmergencyCallData.Comment+xml"

##### 4.5.1. Comment

Data Element: EmergencyCallData.Comment

Use: Optional

XML Element: <Comment>

Description: Human readable text providing additional information to the PSAP staff.

Reason for Need: Explanatory information for values in the data structure.

How Used by Call Taker: To interpret the data provided.

##### 4.5.2. EmergencyCallData.Comment Example

```
<?xml version="1.0" encoding="UTF-8"?>
<com:EmergencyCallData.Comment
  xmlns:com="urn:ietf:params:xml:ns:EmergencyCallData:Comment">
  <com:DataProviderReference>string0987654321@example.org
  </com:DataProviderReference>
  <com:Comment xml:lang="en">This is an example text.</com:Comment>
</com:EmergencyCallData.Comment>
```

Figure 13: EmergencyCallData.Comment Example.

## 5. Issues with getting new types of data into use

This document describes two mechanisms that allow extension of the kind of data provided with an emergency call: define a new block or define a new service specific additional data URL for the DeviceInfo block (Section 4.3.5). While defining new data types and getting a new device or application to send the new data might be easy, getting PSAPs and responders to actually retrieve the data and use it will be difficult. New mechanism providers should understand that acquiring and using new forms of data usually require software upgrades at the PSAP and/or responders, as well as training of call takers and responders in how to interpret and use the information. Legal and operational review might also be needed. Overwhelming a call taker or responder with too much information is highly discouraged. Thus, the barrier to supporting new data is quite high.

The mechanisms this document describes are meant to encourage development of widely supported, common data formats for classes of devices. If all manufacturers of a class of device use the same format, and the data can be shown to improve outcomes, then PSAPs and responders can be encouraged to upgrade their systems and train their staff to use the data. Variations, however well intentioned, are unlikely to be supported.

Implementers should consider that data from sensor-based devices in some cases might not be useful to call takers or PSAPs (and privacy, liability, or other considerations might preclude the PSAP from accessing or handling the data), but might be of use to responders. Each data item provided with the call in conformance with this document can be accessed by responders or other entities in the emergency services, whether or not the data is accessed by the PSAP.

### 5.1. Choosing between defining a new type of block or new type of device/service-specific additional data

For devices that have device or service specific data, there are two choices to carry it. A new block can be defined, or the device/service-specific additional data URL in the DeviceInfo block can be

used and a new type for it defined. The data passed would likely be the same in either case. Considerations for choosing the mechanism under which to register include:

**Applicability:** Information which will be supported by many kinds of devices or services are more appropriately defined as separate blocks.

**Privacy:** Information sent as a device/service-specific additional data URL in the DeviceInfo block is by reference (not by value), which inherently provides some additional privacy protection (since the requester needs to supply a certificate which is verified by the supplier).

**Size:** Information which can be very large might be better sent in the DeviceInfo block, rather than a new block, so that implementations are unable to send the data by value. Conversely, data which is small might best be sent in a separate block so that it can be sent by value.

**Availability of a server:** Providing the data via the device block requires a server be available from which to retrieve the data. Providing the data via new block allows it to be sent by value.

## 6. Data Transport Mechanisms

This section defines how to convey additional data to an emergency service provider. Two different means are specified: the first uses the call signaling; the second uses the <provided-by> element of a PIDF-LO [RFC4119].

1. First, the ability to embed a Uniform Resource Identifier (URI) in an existing SIP header field, the Call-Info header field, is defined. The URI points to the additional data structure. The Call-Info header field is specified in Section 20.9 of [RFC3261].

This document adds a new compound token starting with the value 'EmergencyCallData' for the Call-Info "purpose" parameter. If the "purpose" parameter is set to a value starting with 'EmergencyCallData', then the Call-Info header field contains either an HTTPS URL pointing to an external resource or a CID (content indirection) URI that allows the data structure to be placed in the body of the SIP message. The "purpose" parameter also indicates the kind of data (by its MIME media subtype) that is available at the URI.

As the data is conveyed using a URI in the SIP signaling, the data itself can reside on an external resource, or can be



contained within the body of the SIP message. When the URI refers to data at an external resource, the data is said to be passed by reference. When the URI refers to data contained within the body of the SIP message, the data is said to be passed by value. A PSAP or emergency responder is able to examine the type of data provided and selectively access the data it is interested in, while forwarding all of it (the values or references) to downstream entities.

To be conveyed in a SIP body, additional data about a call is defined as a series of MIME objects (also referred to as a "block" of data). Each block defined in this document is an XML data structure identified by its MIME media type. (Blocks defined by others can be encoded in XML or not, as identified by their MIME registration.) As usual, whenever more than one MIME part is included in the body of a message, MIME multipart (i.e., 'multipart/mixed') encloses them all.

This document defines a set of XML schemas and MIME media types used for each block defined here. When additional data is passed by value in the SIP signaling, each CID URL points to one block in the body. Multiple URIs are used within a Call-Info header field (or multiple Call-Info header fields) to point to multiple blocks. When additional data is provided by reference (in SIP signaling or the <provided-by> element of a PIDF-LO), each HTTPS URL references one block; the data is retrieved with an HTTPS GET operation, which returns the block as an object (the blocks defined here are returned as XML objects).

2. Second, the ability to embed additional data structures in the <provided-by> element of a PIDF-LO [RFC4119] is defined.

In addition to service providers in the call path, the access network provider generally has similar information that can be valuable to the PSAP. When the access network provider and service provider are separate entities, the access network does not participate in the application layer signaling (and hence cannot add a Call-Info header field to the SIP message), but can provide location information in a PIDF-LO. When the access network provider supplies location information in the form of a PIDF-LO from a location server via a location configuration protocol, it has the ability to add the data structures defined in this document (or references to them) within the PIDF-LO.

The data in these data structures is not specific to the location itself, but rather provides descriptive information having to do with the immediate circumstances about the provider's provision of the location (e.g., the identity of the access network

provider, how to contact that entity, what kind of service the access network provides, subscriber information, etc.). This data is similar in nearly every respect to the data known by service providers in the path of the call. The <provided-by> element of the PIDF-LO is a mechanism for the access network provider to supply the information. This document describes a namespace per [RFC4119] for inclusion in the <provided-by> element of a PIDF-LO for adding information known to the access network provider. The access network provider SHOULD provide additional data within a <provided-by> element of a PIDF-LO it returns for emergency use (e.g., if requested with a HELD "responseTime" attribute of "emergencyRouting" or "emergencyDispatch" [RFC5985]).

One or more blocks of data registered in the Emergency Call Additional Data registry, as defined in Section 11.1.9, can be included or referenced in the SIP signaling (using the Call-Info header field) or in the <provided-by> element of a PIDF-LO. For interoperability, only blocks in the registry are permitted to be sent using the mechanisms specified in this document. Since multiple entities are expected to provide sets of data, the data itself needs information describing the source. Consequently, each entity adding additional data MUST supply a "Data Provider" block. All other blocks are optional, but each entity SHOULD supply all blocks where it has at least some of the information in the block.

Note that, as with any mechanism, failures are possible. For example, a block (provided by value or by reference) might not be the type indicated by the "purpose" parameter, or might be badly formed, etc. The general principle that applies to emergency calls is that it is more important for the call to go through than for everything to be correct. Thus, most PSAPs will process a call if at all possible, even if data is missing or other failures occur.

#### 6.1. Transmitting Blocks using Call-Info

A URI to a block MAY be inserted in any SIP request or response method (most often INVITE or MESSAGE), using a Call-Info header field containing a purpose value starting with 'EmergencyCallData', a dot ("."), and the type of data available at the URI. The type of data is denoted by including the root of the MIME media subtype (the 'EmergencyCallData' prefix is not repeated), omitting any suffix such as '+xml'. For example, when referencing a block with MIME media type 'application/EmergencyCallData.ProviderInfo+xml', the 'purpose' parameter is set to 'EmergencyCallData.ProviderInfo'. An example "Call-Info" header field for this would be:

Call-Info: [https://www.example.com/23sedde3;](https://www.example.com/23sedde3;purpose=EmergencyCallData.ProviderInfo)  
purpose="EmergencyCallData.ProviderInfo"

A Call-info header field with a purpose value starting with 'EmergencyCallData' only has meaning in the context of an emergency call (as ascertained by the presence of an emergency service URN in a Request-URI header field of a SIP message), test emergency calls (using an appropriate service URN), and some private-use calls where the endpoints have a preexisting relationship and privacy concerns do not apply because of the relationship; use in other contexts is undefined and is likely to unnecessarily expose confidential data.

If the data is provided by reference, an HTTPS URI MUST be included and consequently Transport Layer Security (TLS) protection is used during the retrieval of the information.

The data can also be supplied by value in any SIP request or response method that is permitted to contain a body (i.e., not a BYE request) [RFC3261]. In this case, Content Indirection (CID) [RFC2392] is used, with the CID URL referencing the MIME body part containing the data. Note that [RFC3261] forbids proxies from altering message bodies, so entities in the call path that add blocks by value need to do so using an appropriate SIP entity (e.g., a back-to-back user agent).

Transmitting data by value is especially useful in certain cases, such as when the data exists in or is generated by the originating device, but is not intended for very large data blocks. Additional security and privacy considerations apply to data transmitted by value, as discussed in Section 9 and Section 10.

More than one Call-Info header field with a purpose value starting with 'EmergencyCallData' can be expected, but at least one MUST be provided. The device MUST provide one unless it knows that a service provider is in the path of the call. The device MAY insert one if it uses a service provider. Each service provider in the path of an emergency call MUST insert its own. For example, a device, a telematics service provider in the call path, as well as the mobile carrier handling the call will each provide one. There might be circumstances where there is a service provider who is unaware that the call is an emergency call and cannot reasonably be expected to determine that it is an emergency call. In that case, that service provider is not expected to provide EmergencyCallData.

When blocks are transmitted by value, the 'purpose' parameter in a Call-Info header field identifies the data, and the CID URL points to the data block in the body (which has a matching Content-ID body part header field). When a data block is carried in a signed or encrypted

body part, the enclosing multipart (e.g., multipart/signed or multipart/encrypted) has the same Content-ID as the data part. This allows an entity to identify and access the data blocks it is interested in without having to dive deeply into the message structure or decrypt parts it is not interested in.

## 6.2. Transmitting Blocks by Reference using the <provided-by> Element

The <EmergencyCallDataReference> element is used to transmit an additional data block by reference within a <provided-by> element of a PIDF-LO. The <EmergencyCallDataReference> element has two attributes: 'ref' to specify the URL, and 'purpose' to indicate the type of data block referenced. The value of 'ref' is an HTTPS URL that resolves to a data structure with information about the call. The value of 'purpose' is the same as used in a 'Call-Info' header field (as specified in Section 6.1).

For example, to reference a block with MIME media type 'application/EmergencyCallData.ProviderInfo+xml', the 'purpose' parameter is set to 'EmergencyCallData.ProviderInfo'. An example <EmergencyCallDataReference> element for this would be:

```
<EmergencyCallDataReference ref="https://www.example.com/23sedde3"
  purpose="EmergencyCallData.ProviderInfo"/>
```

The <EmergencyCallDataReference> element transmits one data block; multiple data blocks are transmitted by using multiple <EmergencyCallDataReference> elements. Multiple <EmergencyCallDataReference> elements MAY be included as child elements inside the <provided-by> element.

The following is a simplified example:

```
<provided-by>
  <EmergencyCallDataReference
    purpose="EmergencyCallData.ServiceInfo"
    ref="https://example.com/ref2" />

  <EmergencyCallDataReference
    purpose="EmergencyCallData.ProviderInfo"
    ref="https://example.com/ref3" />

  <EmergencyCallDataReference
    purpose="EmergencyCallData.Comment"
    ref="https://example.com/ref4" />
</provided-by>
```

#### Example <provided-by> by Reference

For an example in context, Figure 18 shows a PIDF-LO example with an <EmergencyCallDataReference> element pointing to an EmergencyCallData.ServiceInfo data block with the URL in the 'ref' attribute and the purpose attribute set to "EmergencyCallData.ServiceInfo".

### 6.3. Transmitting Blocks by Value using the <provided-by> Element

It is RECOMMENDED that access networks supply the data specified in this document by reference, because PIDF-LOs can be fetched by a client or other entity and stored locally, so providing the data by value risks exposing private information to a larger audience.

The <EmergencyCallDataValue> element is used to transmit one or more additional data blocks by value within a <provided-by> element of a PIDF-LO. Each block being transmitted is placed (as a child element) inside the <EmergencyCallDataValue> element. (The same XML structure as would be contained in the corresponding MIME media type body part is placed inside the <EmergencyCallDataValue> element.) Multiple <EmergencyCallDataValue> elements MAY be included as child elements in the <provided-by> element.

The following is a simplified example:

```
<provided-by>

  <EmergencyCallDataValue>

    <EmergencyCallData.ProviderInfo
      xmlns=
        "urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo">
      <DataProviderReference>flurbit735@es.example.com
        </DataProviderReference>
      <DataProviderString>Access Network Examples, Inc
        </DataProviderString>
      <ProviderID>urn:nena:companyid:Test</ProviderID>
      <ProviderIDSeries>NENA</ProviderIDSeries>
      <TypeOfProvider>Access Network Provider
        </TypeOfProvider>
      <ContactURI>tel:+1-555-555-0897</ContactURI>
      <Language>en</Language>
    </EmergencyCallData.ProviderInfo>

    <EmergencyCallData.Comment
      xmlns=
        "urn:ietf:params:xml:ns:EmergencyCallData:Comment">
      <DataProviderReference>flurbit735@es.example.com
        </DataProviderReference>
      <Comment xml:lang="en">This is an example text.
        </Comment>
    </EmergencyCallData.Comment>

  </EmergencyCallDataValue>

</provided-by>
```

#### Example <provided-by> by Value

For an example in context, Figure 18 shows a PIDF-LO example that contains a <provided-by> element with the <EmergencyCallData.ProviderInfo> and the <EmergencyCallData.Comment> elements as child elements of an <EmergencyCallDataValue> element.

#### 6.4. The Content-Disposition Parameter

RFC 5621 [RFC5621] discusses the handling of message bodies in SIP. It updates and clarifies handling originally defined in RFC 3261 [RFC3261] based on implementation experience. While RFC 3261 did not mandate support for 'multipart' message bodies, 'multipart/mixed' MIME bodies are used by many extensions (including this document)

today. For example, adding a PIDF-LO, SDP, and additional data in body of a SIP message requires a 'multipart' message body.

RFC 3204 [RFC3204] and RFC 3459 [RFC3459] define the 'handling' parameter for the Content-Disposition header field. These RFCs describe how a UAS reacts if it receives a message body whose content type or disposition type it does not understand. If the 'handling' parameter has the value "optional", the UAS ignores the message body. If the 'handling' parameter has the value "required", the UAS returns a 415 (Unsupported Media Type) response. The 'by-reference' disposition type of [RFC5621] allows a SIP message to contain a reference to the body part, and the SIP UA processes the body part according to the reference. This is the case for a Call-info header field containing a Content Indirection (CID) URL.

As an example, a SIP message indicates the Content-Disposition parameter in the body of the SIP message as shown in Figure 14.

```
Content-Type: application/sdp
...Omit Content-Disposition here; defaults are ok

...SDP goes in here

--boundary1
Content-Type: application/pidf+xml
Content-ID: <target123@atlanta.example.com>
Content-Disposition: by-reference;handling=optional

...PIDF-LO goes in here

--boundary1
Content-Type: application/EmergencyCallData.ProviderInfo+xml
Content-ID: <1234567890@atlanta.example.com>
Content-Disposition: by-reference; handling=optional

...Data provider information data goes in here

--boundary1--
```

Figure 14: Example for use of the Content-Disposition Parameter in SIP

## 7. Examples

This section illustrates a longer and more complex example, as shown in Figure 15. In this example additional data is added by the end device, included by the VoIP provider, and provided by the access network provider (via the PIDF-LO).

```

O   +-----+      [=====]      [=====]
/|\  | UA |      [ Access ]      [ VoIP ]
|   +-----+      [ Network ]     [ Provider ]
/\   [ Provider ]     [ example.org ]
      [           ]     [           ]
(1)   [           ] (2)   [           ]
Emergency Call [           ] Emergency Call [           ]
----->
+Device Info   [           ] +Device Info   [           ]
+Data Prov. Info [ ^       ] +Data Provider Info [           ]
+Location URI   [=====.] +Location URI   [=====]
      .
      .
+Location      . [=====]
+Owner/Subscriber Info . [           ] (3)
+Device Info   . (4) [           ]
+Data Provider Info #3 .....> [           ]
      [           ] Emergency Call
      [           ] +Device Info
      [ PSAP       ] +Data Prov. Info #2
      [           ] +Location URI
      [=====]

```

Legend:

--- Emergency Call Setup Procedure  
 ... Location Retrieval/Response

Figure 15: Additional Data Example Flow

The example scenario starts with the end device itself adding device information, owner/subscriber information, a location URI, and data provider information to the outgoing emergency call setup message (see step #1 in Figure 15). The SIP INVITE example is shown in Figure 16.

```

INVITE urn:service:sos SIP/2.0
Via: SIPS/2.0/TLS server.example.com;branch=z9hG4bK74bf9

```



```
Max-Forwards: 70
To: <urn:service:sos>
From: Hannes Tschofenig <sips:hannes@example.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@example.com
Call-Info: <http://www.example.com/hannes/photo.jpg>
           ;purpose=icon,
           <http://www.example.com/hannes/> ;purpose=info,
           <cid:1234567890@atlanta.example.com>
           ;purpose=EmergencyCallData.ProviderInfo,
           <cid:0123456789@atlanta.example.com>
           ;purpose=EmergencyCallData.DeviceInfo
Geolocation: <https://ls.example.net:9768/357yc6s64ceyoiuy5ax3o>
Geolocation-Routing: yes
Accept: application/sdp, application/pidf+xml,
       application/EmergencyCallData.ProviderInfo+xml
CSeq: 31862 INVITE
Contact: <sips:hannes@example.com>
Content-Type: multipart/mixed; boundary=boundary1
Content-Length: ...

--boundary1
Content-Type: application/sdp

...SDP goes here

--boundary1
Content-Type: application/EmergencyCallData.DeviceInfo+xml
Content-ID: <0123456789@atlanta.example.com>
Content-Disposition: by-reference;handling=optional

<?xml version="1.0" encoding="UTF-8"?>
<dev:EmergencyCallData.DeviceInfo
  xmlns:dev=
    "urn:ietf:params:xml:ns:EmergencyCallData:DeviceInfo">
  <dev:DataProviderReference>
    d4b3072df09876543@[93.184.216.119]
  </dev:DataProviderReference>
  <dev:DeviceClassification>laptop</dev:DeviceClassification>
  <dev:UniqueDeviceID
    TypeOfDeviceID="MAC">00-0d-4b-30-72-df
  </dev:UniqueDeviceID>
</dev:EmergencyCallData.DeviceInfo>

--boundary1
Content-Type: application/EmergencyCallData.ProviderInfo+xml
Content-ID: <1234567890@atlanta.example.com>
Content-Disposition: by-reference;handling=optional
```

```
<?xml version="1.0" encoding="UTF-8"?>
<pi:EmergencyCallData.ProviderInfo
  xmlns:pi=
    "urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo">
  <pi:DataProviderReference>d4b3072df09876543@[93.184.216.119]
    </pi:DataProviderReference>
  <pi:DataProviderString>Hannes Tschofenig</pi:DataProviderString>
  <pi:TypeOfProvider>Client</pi:TypeOfProvider>
  <pi:ContactURI>tel:+1-555-555-0123</pi:ContactURI>
  <pi:Language>en</pi:Language>
  <pi:DataProviderContact
    xmlns="urn:ietf:params:xml:ns:vcard-4.0">
    <vcard>
      <fn><text>Hannes Tschofenig</text></fn>
      <n>
        <surname>Hannes</surname>
        <given>Tschofenig</given>
        <additional/>
        <prefix/>
        <suffix>Dipl. Ing.</suffix>
      </n>
      <bday><date>--0203</date></bday>
      <anniversary>
        <date-time>20090808T1430-0500</date-time>
      </anniversary>
      <gender><sex>M</sex></gender>
      <lang>
        <parameters><pref><integer>1</integer></pref>
        </parameters>
        <language-tag>de</language-tag>
      </lang>
      <lang>
        <parameters><pref><integer>2</integer></pref>
        </parameters>
        <language-tag>en</language-tag>
      </lang>
      <adr>
        <parameters>
          <type><text>work</text></type>
          <label><text>Hannes Tschofenig
            Linnoitustie 6
            Espoo, Finland
            02600</text></label>
        </parameters>
        <pobox/>
        <ext/>
        <street>Linnoitustie 6</street>
        <locality>Espoo</locality>
```

```
<region>Uusimaa</region>
<code>02600</code>
<country>Finland</country>
</adr>
<adr>
  <parameters>
    <type><text>home</text></type>
    <label><text>Hannes Tschofenig
      c/o Hotel DuPont
      42 W 11th St
      Wilmington, DE 19801
      USA</text></label>
  </parameters>
  <pobox/>
  <ext/>
  <street>42 W 11th St</street>
    <locality>Wilmington</locality>
    <region>DE</region>
    <code>19801</code>
    <country>USA</country>
</adr>
<tel>
  <parameters>
    <type>
      <text>work</text>
      <text>voice</text>
    </type>
  </parameters>
  <uri>tel:+358 50 4871445</uri>
</tel>
<tel>
  <parameters>
    <type>
      <text>home</text>
      <text>voice</text>
    </type>
  </parameters>
  <uri>tel:+1 555 555 0123</uri>
</tel>
<tel>
  <parameters>
    <type>
      <text>work</text>
      <text>voice</text>
      <text>main-number</text>
    </type>
  </parameters>
  <uri>tel:+1 302 594-3100</uri>
```

```

    </tel>
    <email>
      <parameters><type><text>work</text></type>
      </parameters>
      <text>hannes.tschofenig@nsn.com</text>
    </email>
    <geo>
      <parameters><type><text>work</text></type>
      </parameters>
      <uri>geo:60.210796,24.812924</uri>
    </geo>
    <geo>
      <parameters><type><text>home</text></type>
      </parameters>
      <uri>geo:39.746537,-75.548027</uri>
    </geo>
    <key>
      <parameters>
        <type><text>home</text></type>
      </parameters>
      <uri>https://www.example.com/key.asc</uri>
    </key>
    <tz><text>Finland/Helsinki</text></tz>
    <url>
      <parameters><type><text>home</text></type>
      </parameters>
      <uri>http://example.com/hannes.tschofenig
      </uri>
    </url>
  </vcard>
</pi:DataProviderContact>
</pi:EmergencyCallData.ProviderInfo>
--boundary1--

```

Figure 16: End Device sending SIP INVITE with Additional Data

In this example, information available to the access network provider is included in the call setup message only indirectly via the use of the location reference. The PSAP has to retrieve it via a separate look-up step. Since the access network provider and the VoIP service provider are two independent entities in this scenario, the access network provider is not involved in application layer exchanges; the SIP INVITE transits the access network transparently, as illustrated in steps #1 and #2 (the access network does not alter the SIP INVITE).

The VoIP service provider receives the message and determines, based on the Service URN, that the incoming request is an emergency call.

It performs typical emergency services related tasks (such as location-based routing), and adds additional data, namely service and subscriber information as well as data provider information #2, to the outgoing message. For the example we assume a VoIP service provider that deploys a back-to-back user agent allowing additional data to be included in the body of the SIP message (rather than by reference), which allows us to illustrate the use of multiple data provider info blocks. The resulting message is shown in Figure 17. The SIP INVITE is sent to the PSAP in step #3.

```
INVITE sips:psap@example.org SIP/2.0
Via: SIPS/2.0/TLS server.example.com;branch=z9hG4bK74bf9
Max-Forwards: 70
To: <urn:service:sos>
From: Hannes Tschofenig <sips:hannes@example.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@example.com
Call-Info: <http://www.example.com/hannes/photo.jpg>;
    purpose=icon,
    <http://www.example.com/hannes/>; purpose=info,
    <cid:1234567890@atlanta.example.com>;
    purpose=EmergencyCallData.ProviderInfo
    <cid:0123456789@atlanta.example.com>;
    purpose=EmergencyCallData.DeviceInfo
Call-Info: <cid:bloorpyhex@atlanta.example.com>;
    purpose=EmergencyCallData.ServiceInfo
Call-Info: <cid:aaabbb@atlanta.example.com>;
    purpose=EmergencyCallData.ProviderInfo
Geolocation: <https://ls.example.net:9768/357yc6s64ceyoiuy5ax3o>
Geolocation-Routing: yes
Accept: application/sdp, application/pidf+xml,
    application/EmergencyCallData.ProviderInfo+xml
CSeq: 31862 INVITE
Contact: <sips:hannes@example.com>
Content-Type: multipart/mixed; boundary=boundary1
Content-Length: ...

--boundary1
Content-Type: application/sdp

...SDP goes here

--boundary1
Content-Type: application/EmergencyCallData.DeviceInfo+xml
Content-ID: <0123456789@atlanta.example.com>
Content-Disposition: by-reference;handling=optional

<?xml version="1.0" encoding="UTF-8"?>
```

```

<dev:EmergencyCallData.DeviceInfo
  xmlns:dev=
    "urn:ietf:params:xml:ns:EmergencyCallData:DeviceInfo">
  <dev:DataProviderReference>d4b3072df09876543@[93.184.216.119]
</dev:DataProviderReference>
  <dev:DeviceClassification>laptop</dev:DeviceClassification>
  <dev:UniqueDeviceID
    TypeOfDeviceID="MAC">00-0d-4b-30-72-df</dev:UniqueDeviceID>
</dev:EmergencyCallData.DeviceInfo>

```

```
--boundary1
```

```
Content-Type: application/EmergencyCallData.ProviderInfo+xml
```

```
Content-ID: <1234567890@atlanta.example.com>
```

```
Content-Disposition: by-reference;handling=optional
```

```

<?xml version="1.0" encoding="UTF-8"?>
<pi:EmergencyCallData.ProviderInfo
  xmlns:pi=
    "urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo">
  <pi:DataProviderReference>d4b3072df09876543@[93.184.216.119]
</pi:DataProviderReference>
  <pi:DataProviderString>Hannes Tschofenig
</pi:DataProviderString>
  <pi:TypeOfProvider>Client</pi:TypeOfProvider>
  <pi:ContactURI>tel:+1-555-555-0123</pi:ContactURI>
  <pi:Language>en</pi:Language>
  <pi:DataProviderContact
    xmlns="urn:ietf:params:xml:ns:vcard-4.0">
    <vcard>
      <fn><text>Hannes Tschofenig</text></fn>
      <n>
        <surname>Hannes</surname>
        <given>Tschofenig</given>
        <additional/>
        <prefix/>
        <suffix>Dipl. Ing.</suffix>
      </n>
      <bday><date>--0203</date></bday>
      <anniversary>
        <date-time>20090808T1430-0500</date-time>
      </anniversary>
      <gender><sex>M</sex></gender>
      <lang>
        <parameters><pref><integer>1</integer></pref>
        </parameters>
        <language-tag>de</language-tag>
      </lang>
    </vcard>
  </pi:DataProviderContact>

```

```
<parameters><pref><integer>2</integer></pref>
</parameters>
<language-tag>en</language-tag>
</lang>
<adr>
  <parameters>
    <type><text>work</text></type>
    <label><text>Hannes Tschofenig
      Linnoitustie 6
      Espoo, Finland
      02600</text></label>
  </parameters>
  <pobox/>
  <ext/>
  <street>Linnoitustie 6</street>
  <locality>Espoo</locality>
  <region>Uusimaa</region>
  <code>02600</code>
  <country>Finland</country>
</adr>
<adr>
  <parameters>
    <type><text>home</text></type>
    <label><text>Hannes Tschofenig
      c/o Hotel DuPont
      42 W 11th St
      Wilmington, DE 19801
      USA</text></label>
  </parameters>
  <pobox/>
  <ext/>
  <street>42 W 11th St</street>
  <locality>Wilmington</locality>
  <region>DE</region>
  <code>19801</code>
  <country>USA</country>
</adr>
<tel>
  <parameters>
    <type>
      <text>work</text>
      <text>voice</text>
    </type>
  </parameters>
  <uri>tel:+358 50 4871445</uri>
</tel>
<tel>
  <parameters>
```

```

        <type>
            <text>home</text>
            <text>voice</text>
        </type>
    </parameters>
    <uri>tel:+1 555 555 0123</uri>
</tel>
<email>
    <parameters><type><text>work</text></type>
    </parameters>
    <text>hannes.tschofenig@nsn.com</text>
</email>
<geo>
    <parameters><type><text>work</text></type>
    </parameters>
    <uri>geo:60.210796,24.812924</uri>
</geo>
<geo>
    <parameters><type><text>home</text></type>
    </parameters>
    <uri>geo:39.746537,-75.548027</uri>
</geo>
<key>
    <parameters>
        <type><text>home</text></type>
    </parameters>
    <uri>https://www.example.com/key.asc</uri>
</key>
<tz><text>Finland/Helsinki</text></tz>
<url>
    <parameters><type><text>home</text></type>
    </parameters>
    <uri>http://example.com/hannes.tschofenig</uri>
</url>
</vcard>
</pi:DataProviderContact>
</pi:EmergencyCallData.ProviderInfo>

```

--boundary1

Content-Type: application/EmergencyCallData.ServiceInfo+xml

Content-ID: <bloorpyhex@atlanta.example.com>

Content-Disposition: by-reference;handling=optional

```

<?xml version="1.0" encoding="UTF-8"?>
<svc:EmergencyCallData.ServiceInfo
  xmlns:svc=
    "urn:ietf:params:xml:ns:EmergencyCallData:ServiceInfo">
  <svc:DataProviderReference>string0987654321@example.org

```



```

    </svc:DataProviderReference>
    <svc:ServiceEnvironment>Residence</svc:ServiceEnvironment>
    <svc:ServiceType>VOIP</svc:ServiceType>
    <svc:ServiceMobility>Unknown</svc:ServiceMobility>
  </svc:EmergencyCallData.ServiceInfo>

--boundary1
Content-Type: application/EmergencyCallData.ProviderInfo+xml
Content-ID: <aaabbb@atlanta.example.com>
Content-Disposition: by-reference;handling=optional

<?xml version="1.0" encoding="UTF-8"?>
<pi:EmergencyCallData.ProviderInfo
  xmlns:pi=
    "urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo">
  <pi:DataProviderReference>string0987654321@example.org
  </pi:DataProviderReference>
  <pi:DataProviderString>Exemplar VoIP Provider
  </pi:DataProviderString>
  <pi:ProviderID>urn:nena:companyid:ID123</pi:ProviderID>
  <pi:ProviderIDSeries>NENA</pi:ProviderIDSeries>
  <pi:TypeOfProvider>Service Provider</pi:TypeOfProvider>
  <pi:ContactURI>sip:voip-provider@example.com</pi:ContactURI>
  <pi:Language>en</pi:Language>
  <pi:DataProviderContact
    xmlns="urn:ietf:params:xml:ns:vcard-4.0">
    <vcard>
      <fn><text>John Doe</text></fn>
      <n>
        <surname>John</surname>
        <given>Doe</given>
        <additional/>
        <prefix/>
        <suffix/>
      </n>
      <bday><date>--0203</date></bday>
      <anniversary>
        <date-time>20090808T1430-0500</date-time>
      </anniversary>
      <gender><sex>M</sex></gender>
      <lang>
        <parameters><pref><integer>1</integer></pref>
        </parameters>
        <language-tag>en</language-tag>
      </lang>
      <org>
        <parameters><type><text>work</text></type>
        </parameters>

```

```
        <text>Exemplar VoIP Provider</text>
    </org>
    <adr>
        <parameters>
            <type><text>work</text></type>
            <label><text>John Doe
                123 Middle Street
                The Sticks, IA 50055</text></label>
        </parameters>
        <pobox/>
        <ext/>
        <street>123 Middle Street</street>
        <locality>the Sticks</locality>
        <region>IA</region>
        <code>50055</code>
        <country>USA</country>
    </adr>
    <tel>
        <parameters>
            <type>
                <text>work</text>
                <text>voice</text>
                <text>main-number</text>
            </type>
        </parameters>
        <uri>sips:john.doe@example.com</uri>
    </tel>
    <email>
        <parameters><type><text>work</text></type>
        </parameters>
        <text>john.doe@example.com</text>
    </email>
    <geo>
        <parameters><type><text>work</text></type>
        </parameters>
        <uri>geo:41.761838,-92.963268</uri>
    </geo>
    <tz><text>America/Chicago</text></tz>
    <url>
        <parameters><type><text>home</text></type>
        </parameters>
        <uri>http://www.example.com/john.doe</uri>
    </url>
</vcard>
</pi:DataProviderContact>
</pi:EmergencyCallData.ProviderInfo>
--boundary1--
```

Figure 17: VoIP Provider sending SIP INVITE with Additional Data

Finally, the PSAP requests location information from the access network provider. The response is shown in Figure 18. Along with the location information, additional data is provided in the <provided-by> element of the PIDF-LO. This request and response is step #4.

```
<?xml version="1.0" encoding="UTF-8"?>
<presence xmlns="urn:ietf:params:xml:ns:pidf"
  xmlns:gp="urn:ietf:params:xml:ns:pidf:geopriv10"
  xmlns:gbp="urn:ietf:params:xml:ns:pidf:geopriv10:basicPolicy"
  xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
  entity="pres:alice@atlanta.example.com">
  <dm:device id="target123-1">
    <gp:geopriv>
      <gp:location-info>
        <civicAddress
          xmlns="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr">
          <country>US</country>
          <A1>DE</A1>
          <A3>Wilmington</A3>
          <PRD>W</PRD>
          <RD>11th</RD>
          <STS>Street</STS>
          <HNO>42</HNO>
          <NAM>The Hotel DuPont</NAM>
          <PC>19801</PC>
        </civicAddress>
      </gp:location-info>
      <gp:usage-rules>
        <gbp:retransmission-allowed>true
      </gbp:retransmission-allowed>
        <gbp:retention-expiry>2013-12-10T20:00:00Z
      </gbp:retention-expiry>
      </gp:usage-rules>
      <gp:method>802.11</gp:method>

      <gp:provided-by
        xmlns="urn:ietf:params:xml:ns:EmergencyCallData">

        <EmergencyCallDataReference
          purpose="EmergencyCallData.ServiceInfo"
          ref="https://example.com/ref2" />

        <EmergencyCallDataValue>
          <EmergencyCallData.ProviderInfo
```

```

xmlns=
"urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo">
<DataProviderReference>88QV4FpfZ976T@example.com
</DataProviderReference>
<DataProviderString>Diamond State Exemplar
</DataProviderString>
<ProviderID>urn:nena:companyid:diamond</ProviderID>
<ProviderIDSeries>NENA</ProviderIDSeries>
<TypeOfProvider>Access Network Provider</TypeOfProvider>
<ContactURI>tel:+1-302-555-0000</ContactURI>
<Language>en</Language>
</EmergencyCallData.ProviderInfo>

<EmergencyCallData.Comment
  xmlns="urn:ietf:params:xml:ns:EmergencyCallData:Comment">
  <DataProviderReference>88QV4FpfZ976T@example.com
  </DataProviderReference>
  <Comment xml:lang="en">This is an example text.</Comment>
</EmergencyCallData.Comment>

</EmergencyCallDataValue>
</gp:provided-by>

</gp:geopriv>
<dm:deviceID>mac:00-0d-4b-30-72-df</dm:deviceID>
<dm:timestamp>2013-07-09T20:57:29Z</dm:timestamp>
</dm:device>
</presence>

```

Figure 18: Access Network Provider returning PIDF-LO with Additional Data

## 8. XML Schemas

This section defines the XML schemas of the five data blocks. Additionally, the provided-by schema is specified.

### 8.1. EmergencyCallData.ProviderInfo XML Schema

```

<?xml version="1.0"?>
<xs:schema
  targetNamespace=
    "urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:pi="urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"

```

```
xmlns:xc="urn:ietf:params:xml:ns:vcard-4.0"
elementFormDefault="qualified"
attributeFormDefault="unqualified">

<xs:import namespace="http://www.w3.org/XML/1998/namespace"
  schemaLocation="http://www.w3.org/2009/01/xml.xsd"/>

<xs:import namespace="urn:ietf:params:xml:ns:vcard-4.0"
  schemaLocation="vcard.xsd"/>

<xs:element
  name="EmergencyCallData.ProviderInfo"
  type="pi:ProviderInfoType"/>

<xs:simpleType name="SubcontractorPriorityType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="sub"/>
    <xs:enumeration value="main"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="ProviderInfoType">
  <xs:sequence>
    <xs:element name="DataProviderReference"
      type="xs:token" minOccurs="1" maxOccurs="1"/>

    <xs:element name="DataProviderString"
      type="xs:string" minOccurs="1" maxOccurs="1"/>

    <xs:element name="ProviderID"
      type="xs:string" minOccurs="0" maxOccurs="1"/>

    <xs:element name="ProviderIDSeries"
      type="xs:string" minOccurs="0" maxOccurs="1"/>

    <xs:element name="TypeOfProvider"
      type="xs:token" minOccurs="1" maxOccurs="1"/>

    <xs:element name="ContactURI" type="xs:anyURI"
      minOccurs="1" maxOccurs="1"/>

    <xs:element name="Language" minOccurs="1" maxOccurs="unbounded">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:pattern
value="([a-z]{2,3}((-[a-z]{3}){0,3})?[a-z]{4,8})
(-[a-z]{4})?((-[a-z]{2}|\d{3}))?(-([0-9a-z]{5,8}|
```

```

\d[0-9a-z]{3}))*(-[0-9a-wyz](-[0-9a-z]{2,8}))+*
(-x(-[0-9a-z]{1,8}))+)?|x(-[0-9a-z]{1,8})+|[a-z]{1,3}
(-[0-9a-z]{2,8}){1,2}"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

  <xs:element name="DataProviderContact"
    minOccurs="0" maxOccurs="1">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0"
          maxOccurs="unbounded" ref="xc:vcard"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:element name="SubcontractorPrincipal"
    type="xs:string" minOccurs="0" maxOccurs="1"/>

  <xs:element name="SubcontractorPriority"
    type="pi:SubcontractorPriorityType"
    minOccurs="0" maxOccurs="1"/>

  <xs:any namespace="##other" processContents="lax"
    minOccurs="0" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>

</xs:schema>

```

Figure 19: EmergencyCallData.ProviderInfo XML Schema.

## 8.2. EmergencyCallData.ServiceInfo XML Schema

```

<?xml version="1.0"?>
<xs:schema
  targetNamespace=
    "urn:ietf:params:xml:ns:EmergencyCallData:ServiceInfo"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:svc="urn:ietf:params:xml:ns:EmergencyCallData:ServiceInfo"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd"/>

  <xs:element name="EmergencyCallData.ServiceInfo"
    type="svc:ServiceInfoType"/>

  <xs:complexType name="ServiceInfoType">
    <xs:sequence>
      <xs:element name="DataProviderReference"
        type="xs:token" minOccurs="1" maxOccurs="1"/>

      <xs:element name="ServiceEnvironment"
        type="xs:string" minOccurs="0" maxOccurs="1"/>

      <xs:element name="ServiceType"
        type="xs:string" minOccurs="1"
        maxOccurs="unbounded"/>

      <xs:element name="ServiceMobility"
        type="xs:string" minOccurs="1" maxOccurs="1"/>

      <xs:any namespace="##other" processContents="lax"
        minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

</xs:schema>

```

Figure 20: EmergencyCallData.ServiceInfo XML Schema.

### 8.3. EmergencyCallData.DeviceInfo XML Schema

```

<?xml version="1.0"?>
<xs:schema
  targetNamespace=
    "urn:ietf:params:xml:ns:EmergencyCallData:DeviceInfo"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"

```

```
xmlns:dev="urn:ietf:params:xml:ns:EmergencyCallData:DeviceInfo"
xmlns:xml="http://www.w3.org/XML/1998/namespace"
elementFormDefault="qualified"
attributeFormDefault="unqualified">

<xs:import namespace="http://www.w3.org/XML/1998/namespace"
  schemaLocation="http://www.w3.org/2001/xml.xsd"/>

<xs:element name="EmergencyCallData.DeviceInfo"
  type="dev:DeviceInfoType"/>

<xs:complexType name="DeviceInfoType">
  <xs:sequence>
    <xs:element name="DataProviderReference"
      type="xs:token" minOccurs="1" maxOccurs="1"/>

    <xs:element name="DeviceClassification"
      type="xs:string" minOccurs="0" maxOccurs="1"/>

    <xs:element name="DeviceMfgr"
      type="xs:string" minOccurs="0" maxOccurs="1"/>

    <xs:element name="DeviceModelNr"
      type="xs:string" minOccurs="0" maxOccurs="1"/>

    <xs:element name="UniqueDeviceID" minOccurs="0"
      maxOccurs="unbounded">
      <xs:complexType>
        <xs:simpleContent>
          <xs:extension base="xs:string">
            <xs:attribute name="TypeOfDeviceID"
              type="xs:string"
              use="required"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
    </xs:element>

    <xs:element name="DeviceSpecificData"
      type="xs:anyURI" minOccurs="0" maxOccurs="1"/>

    <xs:element name="DeviceSpecificType"
      type="xs:string" minOccurs="0" maxOccurs="1"/>

    <xs:any namespace="##other" processContents="lax"
      minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
```



```
</xs:schema>
```

Figure 21: EmergencyCallData.DeviceInfo XML Schema.

#### 8.4. EmergencyCallData.SubscriberInfo XML Schema

```
<?xml version="1.0"?>
<xs:schema
  targetNamespace=
    "urn:ietf:params:xml:ns:EmergencyCallData:SubscriberInfo"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:sub=
    "urn:ietf:params:xml:ns:EmergencyCallData:SubscriberInfo"
  xmlns:xc="urn:ietf:params:xml:ns:vcard-4.0"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd"/>

  <xs:import namespace="urn:ietf:params:xml:ns:vcard-4.0"
    schemaLocation="vcard.xsd"/>

  <xs:element name="EmergencyCallData.SubscriberInfo"
    type="sub:SubscriberInfoType"/>

  <xs:complexType name="SubscriberInfoType">
    <xs:complexContent>
      <xs:restriction base="xs:anyType">
        <xs:sequence>
          <xs:element name="DataProviderReference"
            type="xs:token" minOccurs="1" maxOccurs="1"/>

          <xs:element name="SubscriberData">
            <xs:complexType>
              <xs:sequence>
                <xs:element maxOccurs="unbounded"
                  ref="xc:vcard"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>

          <xs:any namespace="##other" processContents="lax"
            minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
</xs:schema>
```

```
        <xs:attribute name="privacyRequested" type="xs:boolean"
            use="required"/>
    </xs:restriction>
</xs:complexContent>
</xs:complexType>

</xs:schema>
```

Figure 22: EmergencyCallData.SubscriberInfo XML Schema.

#### 8.5. EmergencyCallData.Comment XML Schema

```

<?xml version="1.0"?>
<xs:schema
  targetNamespace=
    "urn:ietf:params:xml:ns:EmergencyCallData:Comment"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:com="urn:ietf:params:xml:ns:EmergencyCallData:Comment"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/xml.xsd"/>

  <xs:element name="EmergencyCallData.Comment"
    type="com:CommentType"/>

  <xs:complexType name="CommentType">
    <xs:sequence>
      <xs:element name="DataProviderReference"
        type="xs:token" minOccurs="1" maxOccurs="1"/>

      <xs:element name="Comment"
        type="com:CommentSubType" minOccurs="0"
        maxOccurs="unbounded"/>

      <xs:any namespace="##other" processContents="lax"
        minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="CommentSubType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="xml:lang"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>

</xs:schema>

```

Figure 23: EmergencyCallData.Comment XML Schema.

## 8.6. provided-by XML Schema

This section defines the provided-by schema.

```

<?xml version="1.0"?>

```

```
<xs:schema
  targetNamespace=
    "urn:ietf:params:xml:ns:EmergencyCallData"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:ad="urn:ietf:params:xml:ns:EmergencyCallData"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  xmlns:pi="urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo"
  xmlns:svc="urn:ietf:params:xml:ns:EmergencyCallData:ServiceInfo"
  xmlns:dev="urn:ietf:params:xml:ns:EmergencyCallData:DeviceInfo"
  xmlns:sub=
    "urn:ietf:params:xml:ns:EmergencyCallData:SubscriberInfo"
  xmlns:com="urn:ietf:params:xml:ns:EmergencyCallData:Comment"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <xs:import
    namespace="urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo"
    schemaLocation="ProviderInfo.xsd"/>
  <xs:import
    namespace="urn:ietf:params:xml:ns:EmergencyCallData:ServiceInfo"
    schemaLocation="ServiceInfo.xsd"/>
  <xs:import
    namespace="urn:ietf:params:xml:ns:EmergencyCallData:DeviceInfo"
    schemaLocation="DeviceInfo.xsd"/>
  <xs:import
    namespace=
      "urn:ietf:params:xml:ns:EmergencyCallData:SubscriberInfo"
    schemaLocation="SubscriberInfo.xsd"/>
  <xs:import
    namespace="urn:ietf:params:xml:ns:EmergencyCallData:Comment"
    schemaLocation="Comment.xsd"/>

  <xs:element name="EmergencyCallDataReference"
    type="ad:ByRefType"/>

  <xs:element name="EmergencyCallDataValue"
    type="ad:EmergencyCallDataValueType"/>

  <!-- Additional Data By Reference -->

  <xs:complexType name="ByRefType">
    <xs:complexContent>
      <xs:restriction base="xs:anyType">
        <xs:sequence>
          <xs:any namespace="##other" minOccurs="0"
            maxOccurs="unbounded" processContents="lax"/>
        </xs:sequence>
        <xs:attribute name="purpose" type="xs:token"/>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>
```

```

        use="required"/>
        <xs:attribute name="ref" type="xs:anyURI"
            use="required"/>
    </xs:restriction>
</xs:complexContent>
</xs:complexType>

<!-- Additional Data By Value -->

<xs:complexType name="EmergencyCallDataValueType">
    <xs:sequence>
        <xs:element name="EmergencyCallData.ProviderInfo"
            type="pi:ProviderInfoType"
            minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="EmergencyCallData.ServiceInfo"
            type="svc:ServiceInfoType"
            minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="EmergencyCallData.DeviceInfo"
            type="dev:DeviceInfoType"
            minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="EmergencyCallData.SubscriberInfo"
            type="sub:SubscriberInfoType"
            minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="EmergencyCallData.Comment"
            type="com:CommentType"
            minOccurs="0" maxOccurs="unbounded"/>

        <xs:any namespace="##other" processContents="lax"
            minOccurs="0" maxOccurs="unbounded"/>

    </xs:sequence>
</xs:complexType>

</xs:schema>

```

Figure 24: provided-by XML Schema

## 9. Security Considerations

The data structures described in this document contain information usually considered private. When information is provided by value, entities that are a party to the SIP signaling (such as proxy servers and back-to-back user agents) will have access to it and need to protect it against inappropriate disclosure. An entity that is able to eavesdrop on the SIP signaling will also have access. Some Internet access types (such as in-the-clear Wi-Fi) are more vulnerable than others (such as 3G or 4G cellular data traffic) to eavesdropping. Mechanisms that protect against eavesdropping (such

as Transport Layer Security (TLS) version 1.2 or later) SHOULD be preferentially used whenever feasible. (This requirement is not a "MUST" because there is an existing deployed base of clear-text SIP, and also because, as an emergency call, it is more important for the call to go through than for it to be protected; e.g., the call MUST proceed even if the TLS negotiation or certificate verification fails for whatever reason.) When information is provided by reference, TLS mutual authentication is REQUIRED. That is, HTTPS is REQUIRED for dereferencing, the requestor MUST use a client certificate to authenticate the HTTP request, and the provider of the information is REQUIRED to validate the credentials provided by the requester. While the creation of a public key infrastructure (PKI) that has global scope might be difficult, the alternatives to creating devices and services that can provide critical information securely are more daunting. The provider of the information MAY enforce any policy it wishes to use, but PSAPs and responder agencies are strongly advised to deploy a PKI so that providers of additional data can check the certificate of the client (the requester) and decide the appropriate policy to enforce based on that certificate.

TLS MUST be version 1.2 or later. TLS MUST be version 1.2 or later. It is RECOMMENDED to use only cipher suites that offer Perfect Forward Secrecy (PFS) and avoid Cipher Block Chaining (CBC), and to follow the recommendations in BCP 195 [RFC7525].

Ideally, the PSAP and emergency responders will be given credentials signed by an authority trusted by the data provider. In most circumstances, nationally recognized credentials are sufficient; the emergency services community within a country can arrange a PKI, data providers can be provisioned with the root CA public key for the country. Some nations are developing a PKI for this, and related, purposes. Since calls could be made from devices where the device and/or the service provider(s) are not local to the emergency services authorities, globally recognized credentials are useful. This might be accomplished by extending the notion of the "forest guide" described in [RFC5582] to allow the forest guide to provide the credential of the PKI root for areas for which it has coverage information, but standards for such a mechanism are not yet available. In its absence, the data provider needs to obtain by out of band means the root CA credentials for any areas to which it is willing to provide additional data. With the credential of the root CA for a national emergency services PKI, the data provider server can validate the credentials of an entity requesting additional data by reference.

The data provider also needs a credential that can be verified by the emergency services to know that it is receiving data from an authorized server. The emergency services authorities could provide

credentials, distinguishable from credentials provided to emergency responders and PSAPs, which could be used to validate data providers. Such credentials would have to be acceptable to any PSAP or responder that could receive a call with additional data supplied by that provider. This would be extensible to global credential validation using the forest guide as mentioned above. In the absence of such credentials, the emergency services authorities could maintain a list of local data providers' credentials as provided to them out of band. At a minimum, the emergency services authorities could obtain a credential from the DNS entry of the domain in the Additional Data URI (e.g., using DANE [RFC6698]) to at least validate that the server is known to the domain providing the URI.

Data provided by devices by reference have similar credential validation issues as for service providers, and while the solutions are the same, the challenges of doing so for every device are obviously more difficult, especially when considering root certificate updates, revocation lists, etc. However, in general, devices are not expected to provide data directly by reference, but rather, to either provide data by value, or upload the data to a server which can more reliably make it available and more easily enforce security policy. Devices which do provide data directly by reference, which might include fixed-location sensors, will need to be capable of handling this.

Neither service providers nor devices will supply private information unless the call is recognized as an emergency call. In cellular telephony systems (such as those using 3GPP IMS), there are different procedures for an originating device to place an emergency versus a normal call. If a call that is really an emergency call is initiated as a normal call and the cellular service provider recognizes this, 3GPP IMS permits the service provider to either accept the call anyway or reject it with a specific code that instructs the device to retry the call as an emergency call. Service providers ought to choose the latter, because otherwise the device will not have included the information specified in this document (since the device didn't recognize the call as being an emergency call).

## 10. Privacy Considerations

This document enables functionality for conveying additional information about the caller and the caller's device and service to the callee. Some of this information is personal data and therefore privacy concerns arise. An explicit privacy indicator for information directly relating to the caller's identity is defined and use is mandatory. However, observance of this request for privacy and which information it relates to is determined by the destination

jurisdiction (which replicates functionality provided in some legacy emergency services systems).

There are a number of privacy concerns with non-emergency real-time communication services that are also applicable to emergency calling. Data protection regulation world-wide has, however, decided to create exceptions for emergency services since the drawbacks of disclosing personal data are outweighed by the benefit for the emergency caller. Hence, the data protection rights of individuals are commonly waived for emergency situations. There are, however, still various countries that offer some degree of anonymity for the caller towards PSAP call takers.

The functionality defined in this document far exceeds the amount of information sharing available in the legacy POTS system. For this reason there are additional privacy threats to consider, which are described in more detail in [RFC6973].

**Stored Data Compromise:** There is an increased risk of stored data compromise since additional data is collected and stored in databases. Without adequate measures to secure stored data from unauthorized or inappropriate access at access network providers, service providers, end devices, as well as PSAPs, individuals are exposed to potential financial, reputational, or physical harm.

**Misattribution:** If the personal data collected and conveyed is incorrect or inaccurate then this can lead to misattribution. Misattribution occurs when data or communications related to one individual are attributed to another.

**Identification:** By the nature of the additional data and its capability to provide much richer information about the caller, the call, and the location, the calling party is identified in a much better way. Some users could feel uncomfortable with this degree of information sharing even in emergency services situations.

**Secondary Use:** There is a risk of secondary use, which is the use of collected information about an individual without the individual's consent for a purpose different from that for which the information was collected. The stated purpose of the additional data is for emergency services purposes, but theoretically the same information could be used for any other call as well. Additionally, parties involved in the emergency call could retain the obtained information and re-use it for other, non-emergency services purposes. While technical measures are not in place to prevent such secondary re-use, policy, legal, regulatory, and other non-technical approaches can be effective.



Disclosure: When the data defined in this document is not properly protected (while in transit with traditional communication security techniques, and while stored using access control mechanisms) there is the risk of disclosure, which is the revelation of private information about an individual.

To mitigate these privacy risks the following countermeasures can be taken:

In regions where callers can elect to suppress certain personally identifying information, network or PSAP functionality can inspect privacy flags within the SIP headers to determine what information can be passed, stored, or displayed to comply with local policy or law. RFC 3325 [RFC3325] defines the "id" priv-value token. The presence of this privacy type in a Privacy header field indicates that the user would like the network asserted identity to be kept private with respect to SIP entities outside the trust domain with which the user authenticated, including the PSAP.

This document defines various data structures that contain privacy-sensitive data. For example, identifiers for the device (e.g., serial number, MAC address) or account/SIM (e.g., IMSI), contact information for the user, location of the caller. Local regulations may govern which data is provided in emergency calls, but in general, the emergency call system is aided by the information described in this document. There is a tradeoff between the privacy considerations and the utility of the data. For protection, this specification requires all retrieval of data passed by reference to be protected against eavesdropping and alteration via communication security techniques (namely TLS). Furthermore, security safeguards are required to prevent unauthorized access to stored data. Various security incidents over at least the past few decades have shown that data breaches are not uncommon and are often caused by lack of proper access control frameworks, software bugs (such as buffer overflows), or missing input parsing (such as SQL injection attacks). The risks of data breaches is increased with the obligation for emergency services to retain emergency call related data for extended periods (e.g., several years are the norm).

Finally, it is also worth highlighting the nature of the SIP communication architecture, which introduces additional complications for privacy. Some forms of data can be sent by value in the SIP signaling or by reference (a URL in the SIP signaling). When data is sent by value, all intermediaries have access to the data. As such, these intermediaries could also introduce additional privacy risk. Therefore, in situations where the conveyed information is privacy-sensitive and intermediaries are involved, transmitting by reference might be appropriate, assuming the source of the data can operate a

sufficient dereferencing infrastructure and that proper access control policies are available for distinguishing the different entities dereferencing the reference. Without access control policies any party in possession of the reference is able to resolve the reference and to obtain the data, including intermediaries.

## 11. IANA Considerations

### 11.1. Emergency Call Additional Data Registry

This document creates a new registry called 'Emergency Call Additional Data' with a number of sub-registries.

For several of the sub-registries, "Expert Review" is the criteria for adding new entries. As discussed in Section 5, it can be counterproductive to register new types of data, and as discussed in Section 10, data sent as part of an emergency call can be very privacy-sensitive. In some cases, it is anticipated that various standards bodies dealing with emergency services might need to register new values, and in those cases text below advises the designed expert to verify that the entity requesting the registration is relevant (e.g., a recognized emergency services related SDO). In other cases, especially those where the trade-off between the potential benefit versus danger of new registrations is more conservative (such as Section 11.1.9), "Specification Required" is the criteria, which is a higher hurdle and also implicitly includes an expert review.

The following sub-registries are created for this registry.

#### 11.1.1. Provider ID Series Registry

This document creates a new sub-registry called "Provider ID Series". As defined in [RFC5226], this registry operates under "Expert Review" rules. The expert should determine that the entity requesting a new value is a legitimate issuer of service provider IDs suitable for use in Additional Call Data.

Private entities issuing or using internally-generated IDs are encouraged to register here and to ensure that all IDs they issue or use are unique. This guarantees that IDs issued or used by the entity are globally unique and distinguishable from other IDs issued or used by the same or a different entity. (Some organizations, such as NENA, issue IDs that are unique among all IDs they issue, so an entity using a combination of its NENA ID and the fact that it is from NENA is globally unique. Other entities might not have an ID issued by an organization such as NENA, so they are permitted to use their domain name, but if so, it needs to be unique.)

The content of this registry includes:

Name: An identifier to be used in the 'ProviderIDSeries' element.

Source: The full name of the organization issuing the identifiers.

URL: A URL to the organization for further information.

The initial set of values is listed in Figure 1.

#### 11.1.2. Service Environment Registry

This document creates a new sub-registry called "Service Environment". As defined in [RFC5226], this registry operates under "Expert Review" rules. The expert should determine that the entity requesting a new value is relevant for this service element (e.g., a recognized emergency services related SDO), and that the new value is distinct from existing values, and its use is unambiguous.

The content of this registry includes:

Token: The value to be used in the <ServiceEnvironment> element.

Description: A short description of the value.

The initial set of values is listed in Figure 4.

#### 11.1.3. Service Type Registry

This document creates a new sub-registry called "Service Type". As defined in [RFC5226], this registry operates under "Expert Review" rules. The expert should determine that the entity requesting a new value is relevant for this service element (e.g., a recognized emergency services related SDO) and that the requested value is clearly distinct from other values so that there is no ambiguity as to when the value is to be used or which value is to be used.

The content of this registry includes:

Name: The value to be used in the <ServiceType> element.

Description: A short description of the value.

The initial set of values is listed in Figure 5.

#### 11.1.4. Service Mobility Registry

This document creates a new sub-registry called "Service Mobility". As defined in [RFC5226], this registry operates under "Expert Review" rules. The expert should determine that the entity requesting a new value is relevant for this service element (e.g., a recognized emergency services related SDO) and that the requested value is clearly distinct from other values so that there is no ambiguity as to when the value is to be used or which value is to be used.

The content of this registry includes:

Token: The value used in the <ServiceMobility> element.

Description: A short description of the value.

The initial set of values is listed in Figure 6.

#### 11.1.5. Type of Provider Registry

This document creates a new sub-registry called "Type of Provider". As defined in [RFC5226], this registry operates under "Expert Review". The expert should determine that the proposed new value is distinct from existing values and appropriate for use in the <TypeOfServiceProvider> element

The content of this registry includes:

Token: The value used in the <TypeOfProvider> element.

Description: A short description of the type of service provider.

The initial set of values is defined in Figure 2.

#### 11.1.6. Device Classification Registry

This document creates a new sub-registry called 'Device Classification'. As defined in [RFC5226], this registry operates under "Expert Review" rules. The expert should consider whether the proposed class is unique from existing classes and the definition of the class will be clear to implementors and PSAPs/responders.

The content of this registry includes:

Token: Value used in the <DeviceClassification> element.

Description: Short description identifying the device type.

The initial set of values are defined in Figure 8.

#### 11.1.7. Device ID Type Registry

This document creates a new sub-registry called "Device ID Type". As defined in [RFC5226], this registry operates under "Expert Review" rules. The expert should ascertain that the proposed type is well understood, and provides information which PSAPs and responders are able to use to uniquely identify a device. (For example, a biometric fingerprint used to authenticate a device would not normally be useful by a PSAP or responder to identify a device.)

The content of this registry includes:

Token: The value to be placed in the <TypeOfDeviceID> element.

Description: Short description identifying the type of the device ID.

The initial set of values are defined in Figure 9.

#### 11.1.8. Device/Service Data Type Registry

This document creates a new sub-registry called "Device/Service Data Type". As defined in [RFC5226], this registry operates under "Specification Required" rules, which include an explicit expert review. The designated expert should ascertain that the proposed type is well understood, and provides information useful to PSAPs and responders. The specification must contain a complete description of the data, and a precise format specification suitable to allow interoperable implementations.

The content of this registry includes:

Token: The value to be placed in the <DeviceSpecificType> element.

Description: Short description identifying the data.

Specification: Citation for the specification of the data.

The initial set of values are listed in Figure 10.

#### 11.1.9. Emergency Call Data Types Registry

This document creates a new sub-registry called 'Emergency Call Data Types'. As defined in [RFC5226], this registry operates under "Specification Required" rules, which include an explicit expert review. The expert is responsible for verifying that the document

contains a complete and clear specification and the proposed functionality does not obviously duplicate existing functionality. The expert is also responsible for verifying that the block is correctly categorized per the description of the categories in Section 1.

The registry contains an entry for every data block that can be sent with an emergency call using the mechanisms as specified in this document. Each data block is identified by the "root" of its MIME media subtype (which is the part after 'EmergencyCallData.'). If the MIME media subtype does not start with 'EmergencyCallData.', then it cannot be registered here nor used in a Call-Info header field as specified in this document. The subtype MAY exist under any MIME media type (although most commonly these are under 'Application/' this is NOT REQUIRED), however, to be added to the registry the "root" needs to be unique regardless of the MIME media type.

The content of this registry includes:

Token: The root of the data's MIME media subtype (not including the 'EmergencyCallData' prefix and any suffix such as '+xml')

Data About: A hint as to if the block is considered descriptive of the call, the caller, or the location (or is applicable to more than one), which can help PSAPs and other entities determine if they wish to process the block. Note that this is only a hint; entities need to consider the block's contents, not just this field, when determining if they wish to process the block (which is why the field only exists in the registry, and is not contained within the block). The value MUST be either "The Call", "The Caller", "The Location", or "Multiple". New values are created by extending this registry in a subsequent RFC.

Reference: The document that describes the data object

Note that the tokens in this registry are part of the 'EmergencyCallData' compound value; when used as a value of the 'purpose' parameter of a Call-Info header field, the values listed in this registry are prefixed by 'EmergencyCallData.' per the 'EmergencyCallData' registration Section 11.2.

The initial set of values are listed in Figure 25.

Token	Data About	Reference
ProviderInfo	The Call	[This RFC]
ServiceInfo	The Call	[This RFC]
DeviceInfo	The Call	[This RFC]
SubscriberInfo	The Call	[This RFC]
Comment	The Call	[This RFC]

Figure 25: Additional Data Blocks Registry

### 11.2. 'EmergencyCallData' Purpose Parameter Value

This document defines the 'EmergencyCallData' value for the 'purpose' parameter of the Call-Info header field [RFC3261]. IANA has added this document to the list of references for the 'purpose' value of Call-Info in the Header Field Parameters and Parameter Values sub-registry of the Session Initiation Protocol (SIP) Parameters registry. Note that 'EmergencyCallData' is a compound value; when used as a value of the 'purpose' parameter of a Call-Info header field, 'EmergencyCallData' is immediately followed by a dot ('.') and a value from the 'Emergency Call Data Types' registry Section 11.1.9.

### 11.3. URN Sub-Namespace Registration for <provided-by> Registry Entry

This section registers the namespace specified in Section 11.5.1 in the provided-by registry established by RFC 4119, for usage within the <provided-by> element of a PIDF-LO.

The schema for the <provided-by> element used by this document is specified in Section 8.6.

### 11.4. MIME Registrations

#### 11.4.1. MIME Content-type Registration for 'application/ EmergencyCallData.ProviderInfo+xml'

This specification requests the registration of a new MIME media type according to the procedures of RFC 6838 [RFC6838] and guidelines in RFC 7303 [RFC7303].

MIME media type name: application

MIME media subtype name: EmergencyCallData.ProviderInfo+xml

Mandatory parameters: none

Optional parameters: charset (indicates the character encoding of the contents)

Encoding considerations: Uses XML, which can contain 8-bit characters, depending on the character encoding. See Section 3.2 of RFC 7303 [RFC7303].

Security considerations: This content type is designed to carry the data provider information, which is a sub-category of additional data about an emergency call. Since this data can contain personal information, appropriate precautions are needed to limit unauthorized access, inappropriate disclosure, and eavesdropping of personal information. Please refer to Section 9 and Section 10 for more information.

Interoperability considerations: None

Published specification: [TBD: This specification]

Applications which use this media type: Emergency Services

Additional information:

Magic Number: None

File Extension: .xml

Macintosh file type code: 'TEXT'

Person and email address for further information: Hannes Tschofenig, Hannes.Tschofenig@gmx.net

Intended usage: LIMITED USE

Author: This specification is a work item of the IETF ECRIT working group, with mailing list address <ecrit@ietf.org>.

Change controller: The IESG <iesg@ietf.org>

#### 11.4.2. MIME Content-type Registration for 'application/ EmergencyCallData.ServiceInfo+xml'

This specification requests the registration of a new MIME media type according to the procedures of RFC 6838 [RFC6838] and guidelines in RFC 7303 [RFC7303].

MIME media type name: application



MIME media subtype name: EmergencyCallData.ServiceInfo+xml

Mandatory parameters: none

Optional parameters: charset (indicates the character encoding of the contents)

Encoding considerations: Uses XML, which can contain 8-bit characters, depending on the character encoding. See Section 3.2 of RFC 7303 [RFC7303].

Security considerations: This content type is designed to carry the service information, which is a sub-category of additional data about an emergency call. Since this data can contain personal information, appropriate precautions are needed to limit unauthorized access, inappropriate disclosure, and eavesdropping of personal information. Please refer to Section 9 and Section 10 for more information.

Interoperability considerations: None

Published specification: [TBD: This specification]

Applications which use this media type: Emergency Services

Additional information:

    Magic Number: None

    File Extension: .xml

    Macintosh file type code: 'TEXT'

Person and email address for further information: Hannes Tschofenig, Hannes.Tschofenig@gmx.net

Intended usage: LIMITED USE

Author: This specification is a work item of the IETF ECRIT working group, with mailing list address <ecrit@ietf.org>.

Change controller: The IESG <iesg@ietf.org>

#### 11.4.3. MIME Content-type Registration for 'application/ EmergencyCallData.DeviceInfo+xml'

This specification requests the registration of a new MIME media type according to the procedures of RFC 6838 [RFC6838] and guidelines in RFC 7303 [RFC7303].

MIME media type name: application

MIME media subtype name: EmergencyCallData.DeviceInfo+xml

Mandatory parameters: none

Optional parameters: charset (indicates the character encoding of the contents)

Encoding considerations: Uses XML, which can contain 8-bit characters, depending on the character encoding. See Section 3.2 of RFC 7303 [RFC7303].

Security considerations: This content type is designed to carry device information, which is a sub-category of additional data about an emergency call. Since this data contains personal information, appropriate precautions need to be taken to limit unauthorized access, inappropriate disclosure to third parties, and eavesdropping of this information. Please refer to Section 9 and Section 10 for more information.

Interoperability considerations: None

Published specification: [TBD: This specification]

Applications which use this media type: Emergency Services

Additional information:

Magic Number: None

File Extension: .xml

Macintosh file type code: 'TEXT'

Person and email address for further information: Hannes  
Tschofenig, Hannes.Tschofenig@gmx.net

Intended usage: LIMITED USE

Author: This specification is a work item of the IETF ECRIT working group, with mailing list address <ecrit@ietf.org>.

Change controller: The IESG <iesg@ietf.org>

#### 11.4.4. MIME Content-type Registration for 'application/ EmergencyCallData.SubscriberInfo+xml'

This specification requests the registration of a new MIME media type according to the procedures of RFC 6838 [RFC6838] and guidelines in RFC 7303 [RFC7303].

MIME media type name: application

MIME media subtype name: EmergencyCallData.SubscriberInfo+xml

Mandatory parameters: none

Optional parameters: charset (indicates the character encoding of the contents)

Encoding considerations: Uses XML, which can contain 8-bit characters, depending on the character encoding. See Section 3.2 of RFC 7303 [RFC7303].

Security considerations: This content type is designed to carry owner/subscriber information, which is a sub-category of additional data about an emergency call. Since this data contains personal information, appropriate precautions need to be taken to limit unauthorized access, inappropriate disclosure to third parties, and eavesdropping of this information. Please refer to Section 9 and Section 10 for more information.

Interoperability considerations: None

Published specification: [TBD: This specification]

Applications which use this media type: Emergency Services

Additional information:

Magic Number: None

File Extension: .xml

Macintosh file type code: 'TEXT'

Person and email address for further information: Hannes  
Tschofenig, Hannes.Tschofenig@gmx.net

Intended usage: LIMITED USE

Author: This specification is a work item of the IETF ECRIT  
working group, with mailing list address <ecrit@ietf.org>.

Change controller: The IESG <iesg@ietf.org>

#### 11.4.5. MIME Content-type Registration for 'application/ EmergencyCallData.Comment+xml'

This specification requests the registration of a new MIME media type according to the procedures of RFC 6838 [RFC6838] and guidelines in RFC 7303 [RFC7303].

MIME media type name: application

MIME media subtype name: EmergencyCallData.Comment+xml

Mandatory parameters: none

Optional parameters: charset (indicates the character encoding of the contents)

Encoding considerations: Uses XML, which can contain 8-bit characters, depending on the character encoding. See Section 3.2 of RFC 7303 [RFC7303].

Security considerations: This content type is designed to carry a comment, which is a sub-category of additional data about an emergency call. This data can contain personal information. Appropriate precautions are needed to limit unauthorized access, inappropriate disclosure to third parties, and eavesdropping of this information. Please refer to Section 9 and Section 10 for more information.

Interoperability considerations: None

Published specification: [TBD: This specification]

Applications which use this media type: Emergency Services

Additional information:

Magic Number: None

File Extension: .xml

Macintosh file type code: 'TEXT'

Person and email address for further information: Hannes  
Tschofenig, Hannes.Tschofenig@gmx.net

Intended usage: LIMITED USE

Author: This specification is a work item of the IETF ECRIT  
working group, with mailing list address <ecrit@ietf.org>.

Change controller: The IESG <iesg@ietf.org>

## 11.5. URN Sub-Namespace Registration

### 11.5.1. Registration for urn:ietf:params:xml:ns:EmergencyCallData

This section registers a new XML namespace, as per the guidelines in  
RFC 3688 [RFC3688].

URI: urn:ietf:params:xml:ns:EmergencyCallData

Registrant Contact: IETF, ECRIT working group, <ecrit@ietf.org>, as  
delegated by the IESG <iesg@ietf.org>.

XML:

```
BEGIN
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
    "http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <meta http-equiv="content-type"
        content="text/html; charset=iso-8859-1"/>
    <title>Namespace for Additional Emergency Call Data</title>
</head>
<body>
    <h1>Namespace for Additional Data related to an Emergency Call
    </h1>
    <p>See [TBD: This document].</p>
</body>
</html>
END
```

## 11.5.2. Registration for

urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo

This section registers a new XML namespace, as per the guidelines in RFC 3688 [RFC3688].

URI: urn:ietf:params:xml:ns:EmergencyCallData:ProviderInfo

Registrant Contact: IETF, ECRIT working group, <ecrit@ietf.org>, as delegated by the IESG <iesg@ietf.org>.

XML:

```
BEGIN
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
  "http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="content-type"
    content="text/html; charset=iso-8859-1"/>
  <title>Namespace for Additional Emergency Call Data:
    Data Provider Information</title>
</head>
<body>
  <h1>Namespace for Additional Data related to an Emergency Call
    </h1>
  <h2>Data Provider Information</h2>
  <p>See [TBD: This document].</p>
</body>
</html>
END
```

## 11.5.3. Registration for

urn:ietf:params:xml:ns:EmergencyCallData:ServiceInfo

This section registers a new XML namespace, as per the guidelines in RFC 3688 [RFC3688].

URI: urn:ietf:params:xml:ns:EmergencyCallData:ServiceInfo

Registrant Contact: IETF, ECRIT working group, <ecrit@ietf.org>, as delegated by the IESG <iesg@ietf.org>.

XML:

```
BEGIN
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
  "http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="content-type"
    content="text/html; charset=iso-8859-1"/>
  <title>Namespace for Additional Emergency Call Data:
    Service Information</title>
</head>
<body>
  <h1>Namespace for Additional Data related to an Emergency Call
    </h1>
  <h2>Service Information</h2>
  <p>See [TBD: This document].</p>
</body>
</html>
END
```

#### 11.5.4. Registration for

urn:ietf:params:xml:ns:EmergencyCallData:DeviceInfo

This section registers a new XML namespace, as per the guidelines in RFC 3688 [RFC3688].

URI: urn:ietf:params:xml:ns:EmergencyCallData:DeviceInfo

Registrant Contact: IETF, ECRIT working group, <ecrit@ietf.org>, as delegated by the IESG <iesg@ietf.org>.

XML:

```
BEGIN
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
  "http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="content-type"
    content="text/html; charset=iso-8859-1"/>
  <title>Namespace for Additional Emergency Call Data:
    Device Information</title>
</head>
<body>
  <h1>Namespace for Additional Data related to an Emergency Call
    </h1>
  <h2>Device Information</h2>
  <p>See [TBD: This document].</p>
</body>
</html>
END
```

#### 11.5.5. Registration for

urn:ietf:params:xml:ns:EmergencyCallData:SubscriberInfo

This section registers a new XML namespace, as per the guidelines in RFC 3688 [RFC3688].

URI: urn:ietf:params:xml:ns:EmergencyCallData:SubscriberInfo

Registrant Contact: IETF, ECRIT working group, <ecrit@ietf.org>, as delegated by the IESG <iesg@ietf.org>.

XML:



```
BEGIN
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
  "http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="content-type"
    content="text/html; charset=iso-8859-1"/>
  <title>Namespace for Additional Emergency Call Data:
    Owner/Subscriber Information</title>
</head>
<body>
  <h1>Namespace for Additional Data related to an Emergency Call
    </h1>
  <h2> Owner/Subscriber Information</h2>
<p>See [TBD: This document].</p>
</body>
</html>
END
```

#### 11.5.6. Registration for

urn:ietf:params:xml:ns:EmergencyCallData:Comment

This section registers a new XML namespace, as per the guidelines in RFC 3688 [RFC3688].

URI: urn:ietf:params:xml:ns:EmergencyCallData:Comment

Registrant Contact: IETF, ECRIT working group, <ecrit@ietf.org>, as delegated by the IESG <iesg@ietf.org>.

XML:

```
BEGIN
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
  "http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta http-equiv="content-type"
    content="text/html; charset=iso-8859-1"/>
  <title>Namespace for Additional Emergency Call Data:Comment
  </title>
</head>
<body>
  <h1>Namespace for Additional Data related to an Emergency Call
  </h1>
  <h2> Comment</h2>
<p>See [TBD: This document].</p>
</body>
</html>
END
```

#### 11.6. Schema Registrations

This specification registers the following schemas, as per the guidelines in RFC 3688 [RFC3688].

Name: Provided-by Schema

URI: urn:ietf:params:xml:schema:EmergencyCallData

Registrant Contact: IETF, ECRIT Working Group (ecrit@ietf.org), as delegated by the IESG (iesg@ietf.org).

XML: The XML schema can be found in Section 8.6.

Name: ProviderInfo Schema

URI: urn:ietf:params:xml:schema:emergencycalldata:ProviderInfo

Registrant Contact: IETF, ECRIT Working Group (ecrit@ietf.org), as delegated by the IESG (iesg@ietf.org).

XML: The XML schema can be found in Figure 19.

Name: ServiceInfo Schema

URI: urn:ietf:params:xml:schema:emergencycalldata:ServiceInfo

Registrant Contact: IETF, ECRIT Working Group (ecrit@ietf.org), as delegated by the IESG (iesg@ietf.org).

XML: The XML schema can be found in Figure 20.

Name: DeviceInfo Schema

URI: urn:ietf:params:xml:schema:emergencycalldata:DeviceInfo

Registrant Contact: IETF, ECRIT Working Group (ecrit@ietf.org), as delegated by the IESG (iesg@ietf.org).

XML: The XML schema can be found in Figure 21.

Name: SubscriberInfo Schema

URI: urn:ietf:params:xml:schema:emergencycalldata:SubscriberInfo

Registrant Contact: IETF, ECRIT Working Group (ecrit@ietf.org), as delegated by the IESG (iesg@ietf.org).

XML: The XML schema can be found in Section 8.4.

Name: Comment Schema

URI: urn:ietf:params:xml:schema:emergencycalldata:comment

Registrant Contact: IETF, ECRIT Working Group (ecrit@ietf.org), as delegated by the IESG (iesg@ietf.org).

XML: The XML schema can be found in Section 8.5.

Name: Additional Data VCard Schema

URI: urn:ietf:params:xml:ns:vcard-4.0

Registrant Contact: IETF, ECRIT Working Group (ecrit@ietf.org), as delegated by the IESG (iesg@ietf.org).

XML: The XML schema can be found in Appendix A.

#### 11.7. VCard Parameter Value Registration

This document registers a new value in the vCARD Parameter Values registry as defined by [RFC6350] with the following template:

Value: main

Purpose: The main telephone number, typically of an enterprise, as opposed to a direct dial number of an individual employee

Conformance: This value can be used with the "TYPE" parameter applied on the "TEL" property.

Example(s): TEL;VALUE=uri;TYPE="main,voice";PREF=1:tel:+1-418-656-9000

## 12. Acknowledgments

This work was originally started in NENA and has benefited from a large number of participants in NENA standardization efforts, originally in the Long Term Definition Working Group, the Data Technical Committee and most recently the Additional Data working group. The authors are grateful for the initial work and extended comments provided by many NENA participants, including Delaine Arnold, Marc Berryman, Guy Caron, Mark Fletcher, Brian Dupras, James Leyerle, Kathy McMahon, Christian Militeau, Ira Pyles, Matt Serra, and Robert (Bob) Sherry. Amursana Khiyod, Robert Sherry, Frank Rahoi, Scott Ross, and Tom Klepetka provided valuable feedback regarding the vCard/xCard use in this specification.

We would also like to thank Paul Kyzivat, Gunnar Hellstrom, Martin Thomson, Keith Drage, Laura Liess, Chris Santer, Barbara Stark, Chris Santer, Archie Cobbs, Magnus Nystrom, Stephen Farrell, Amanda Baber, Dan Banks, Andrew Newton, Philip Reichl, and Francis Dupont for their review comments. Alissa Cooper, Guy Caron, Ben Campbell, and Barry Leiba deserves special mention for their detailed and extensive review comments, which were very helpful and appreciated.

## 13. References

### 13.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC2392] Levinson, E., "Content-ID and Message-ID Uniform Resource Locators", RFC 2392, DOI 10.17487/RFC2392, August 1998, <<http://www.rfc-editor.org/info/rfc2392>>.
- [RFC3204] Zimmerer, E., Peterson, J., Vemuri, A., Ong, L., Audet, F., Watson, M., and M. Zonoun, "MIME media types for ISUP and QSIG Objects", RFC 3204, DOI 10.17487/RFC3204, December 2001, <<http://www.rfc-editor.org/info/rfc3204>>.

- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <<http://www.rfc-editor.org/info/rfc3261>>.
- [RFC3459] Burger, E., "Critical Content Multi-purpose Internet Mail Extensions (MIME) Parameter", RFC 3459, DOI 10.17487/RFC3459, January 2003, <<http://www.rfc-editor.org/info/rfc3459>>.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <<http://www.rfc-editor.org/info/rfc3688>>.
- [RFC3966] Schulzrinne, H., "The tel URI for Telephone Numbers", RFC 3966, DOI 10.17487/RFC3966, December 2004, <<http://www.rfc-editor.org/info/rfc3966>>.
- [RFC4119] Peterson, J., "A Presence-based GEOPRIV Location Object Format", RFC 4119, DOI 10.17487/RFC4119, December 2005, <<http://www.rfc-editor.org/info/rfc4119>>.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, DOI 10.17487/RFC5226, May 2008, <<http://www.rfc-editor.org/info/rfc5226>>.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, DOI 10.17487/RFC5322, October 2008, <<http://www.rfc-editor.org/info/rfc5322>>.
- [RFC5621] Camarillo, G., "Message Body Handling in the Session Initiation Protocol (SIP)", RFC 5621, DOI 10.17487/RFC5621, September 2009, <<http://www.rfc-editor.org/info/rfc5621>>.
- [RFC5646] Phillips, A., Ed. and M. Davis, Ed., "Tags for Identifying Languages", BCP 47, RFC 5646, DOI 10.17487/RFC5646, September 2009, <<http://www.rfc-editor.org/info/rfc5646>>.
- [RFC6350] Perreault, S., "vCard Format Specification", RFC 6350, DOI 10.17487/RFC6350, August 2011, <<http://www.rfc-editor.org/info/rfc6350>>.
- [RFC6351] Perreault, S., "xCard: vCard XML Representation", RFC 6351, DOI 10.17487/RFC6351, August 2011, <<http://www.rfc-editor.org/info/rfc6351>>.

- [RFC6838] Freed, N., Klensin, J., and T. Hansen, "Media Type Specifications and Registration Procedures", BCP 13, RFC 6838, DOI 10.17487/RFC6838, January 2013, <<http://www.rfc-editor.org/info/rfc6838>>.
- [RFC7303] Thompson, H. and C. Lilley, "XML Media Types", RFC 7303, DOI 10.17487/RFC7303, July 2014, <<http://www.rfc-editor.org/info/rfc7303>>.

### 13.2. Informational References

- [ECRIT-WG-wiki] IETF, "ECRIT WG Wiki", July 2015, <<http://tools.ietf.org/wg/ecrit/trac/attachment/wiki/WikiStart/additional-data-examples.zip>>.
- [I-D.ietf-slim-negotiating-human-language] Gellens, R., "Negotiating Human Language in Real-Time Communications", draft-ietf-slim-negotiating-human-language-01 (work in progress), March 2016.
- [IANA-XML-Schemas] IANA, "IANA XML Schemas", July 2015, <<http://www.iana.org/assignments/xml-registry/xml-registry.xhtml#schema>>.
- [IEEE-1512-2006] IEEE, "1512-2006 - IEEE Standard for Common Incident Management Message Sets for Use by Emergency Management Centers", Jun 2006, <<https://standards.ieee.org/findstds/standard/1512-2006.html>>.
- [LanguageTagRegistry] IANA, "Language Subtag Registry", Feb 2015, <<http://www.iana.org/assignments/language-subtag-registry/language-subtag-registry>>.
- [LERG] Telcordia Technologies, Inc., "Local Exchange Routing Guide (LERG)", ANI II Digits Definitions , June 2015.
- [NANP] North American Numbering Plan Administration, "ANI II Digits Assignments", September 2015, <[http://nanpa.com/number\\_resource\\_info/ani\\_ii\\_assignments.html](http://nanpa.com/number_resource_info/ani_ii_assignments.html)>.
- [nc911] North Carolina 911 Board, "North Carolina Telecommunicator Reference", January 2009, <[https://www.nc911.nc.gov/pdf/A\\_TelecommunicatorReference.pdf](https://www.nc911.nc.gov/pdf/A_TelecommunicatorReference.pdf)>.

- [NENA-02-010] National Emergency Number Association (NENA), "NENA Standard Data Formats for 9-1-1 Data Exchange & GIS Mapping", NENA Standard 02-010, December 2010, <<http://www.nena.org>>.
- [RFC3325] Jennings, C., Peterson, J., and M. Watson, "Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks", RFC 3325, DOI 10.17487/RFC3325, November 2002, <<http://www.rfc-editor.org/info/rfc3325>>.
- [RFC3840] Rosenberg, J., Schulzrinne, H., and P. Kyzivat, "Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)", RFC 3840, DOI 10.17487/RFC3840, August 2004, <<http://www.rfc-editor.org/info/rfc3840>>.
- [RFC5012] Schulzrinne, H. and R. Marshall, Ed., "Requirements for Emergency Context Resolution with Internet Technologies", RFC 5012, DOI 10.17487/RFC5012, January 2008, <<http://www.rfc-editor.org/info/rfc5012>>.
- [RFC5139] Thomson, M. and J. Winterbottom, "Revised Civic Location Format for Presence Information Data Format Location Object (PIDF-LO)", RFC 5139, DOI 10.17487/RFC5139, February 2008, <<http://www.rfc-editor.org/info/rfc5139>>.
- [RFC5491] Winterbottom, J., Thomson, M., and H. Tschofenig, "GEOPRIV Presence Information Data Format Location Object (PIDF-LO) Usage Clarification, Considerations, and Recommendations", RFC 5491, DOI 10.17487/RFC5491, March 2009, <<http://www.rfc-editor.org/info/rfc5491>>.
- [RFC5582] Schulzrinne, H., "Location-to-URL Mapping Architecture and Framework", RFC 5582, DOI 10.17487/RFC5582, September 2009, <<http://www.rfc-editor.org/info/rfc5582>>.
- [RFC5962] Schulzrinne, H., Singh, V., Tschofenig, H., and M. Thomson, "Dynamic Extensions to the Presence Information Data Format Location Object (PIDF-LO)", RFC 5962, DOI 10.17487/RFC5962, September 2010, <<http://www.rfc-editor.org/info/rfc5962>>.
- [RFC5985] Barnes, M., Ed., "HTTP-Enabled Location Delivery (HELD)", RFC 5985, DOI 10.17487/RFC5985, September 2010, <<http://www.rfc-editor.org/info/rfc5985>>.

- [RFC6443] Rosen, B., Schulzrinne, H., Polk, J., and A. Newton, "Framework for Emergency Calling Using Internet Multimedia", RFC 6443, DOI 10.17487/RFC6443, December 2011, <<http://www.rfc-editor.org/info/rfc6443>>.
- [RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <<http://www.rfc-editor.org/info/rfc6698>>.
- [RFC6848] Winterbottom, J., Thomson, M., Barnes, R., Rosen, B., and R. George, "Specifying Civic Address Extensions in the Presence Information Data Format Location Object (PIDF-LO)", RFC 6848, DOI 10.17487/RFC6848, January 2013, <<http://www.rfc-editor.org/info/rfc6848>>.
- [RFC6881] Rosen, B. and J. Polk, "Best Current Practice for Communications Services in Support of Emergency Calling", BCP 181, RFC 6881, DOI 10.17487/RFC6881, March 2013, <<http://www.rfc-editor.org/info/rfc6881>>.
- [RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <<http://www.rfc-editor.org/info/rfc6973>>.
- [RFC7035] Thomson, M., Rosen, B., Stanley, D., Bajko, G., and A. Thomson, "Relative Location Representation", RFC 7035, DOI 10.17487/RFC7035, October 2013, <<http://www.rfc-editor.org/info/rfc7035>>.
- [RFC7090] Schulzrinne, H., Tschofenig, H., Holmberg, C., and M. Patel, "Public Safety Answering Point (PSAP) Callback", RFC 7090, DOI 10.17487/RFC7090, April 2014, <<http://www.rfc-editor.org/info/rfc7090>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<http://www.rfc-editor.org/info/rfc7525>>.

### 13.3. URIs

- [1] <http://www.nena.org/?page=cid2014>
- [2] <http://www.nena.org/?page=CompanyID>



## Appendix A. XML Schema for vCard/xCard

This section contains the vCard/xCard XML schema version of the Relax NG schema defined in RFC 6351 [RFC6351] for use with the XML schemas defined in this document. In addition to mapping the Relax NG schema to an XML schema this specification furthermore applies an errata raised for RFC 6351 regarding the type definition (see RFC Errata ID: 3047).

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  targetNamespace="urn:ietf:params:xml:ns:vcard-4.0"
  xmlns:ns1="urn:ietf:params:xml:ns:vcard-4.0">
  <!--

    3.3
    iana-token = xs:string { pattern = "[a-zA-Z0-9-]+" }
    x-name = xs:string { pattern = "x-[a-zA-Z0-9-]+" }
  -->
  <xs:simpleType name="iana-token">
    <xs:annotation>
      <xs:documentation>vCard Format Specification
    </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <xs:simpleType name="x-name">
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
  <!--

    4.1
  -->
  <xs:element name="text" type="xs:string"/>
  <xs:group name="value-text-list">
    <xs:sequence>
      <xs:element maxOccurs="unbounded" ref="ns1:text"/>
    </xs:sequence>
  </xs:group>
  <!-- 4.2 -->
  <xs:element name="uri" type="xs:anyURI"/>
  <!-- 4.3.1 -->
  <xs:element name="date"
    substitutionGroup="ns1:value-date-and-or-time">
    <xs:simpleType>
      <xs:restriction base="xs:string">
```

```

        <xs:pattern value=
"\d{8}|\d{4}-\d\d|--\d\d(\d\d)?|---\d\d"/>
    </xs:restriction>
</xs:simpleType>
</xs:element>
<!-- 4.3.2 -->
<xs:element name="time"
substitutionGroup="ns1:value-date-and-or-time">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value=
"(\d\d(\d\d(\d\d)?)|-\d\d(\d\d)?|--\d\d)(Z|[\+-]\d\d(\d\d)?)" />
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- 4.3.3 -->
<xs:element name="date-time"
substitutionGroup="ns1:value-date-and-or-time">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value=
"(\d{8}|--\d{4}|---\d\d)T\d\d(\d\d(\d\d)?)(Z|[\+-]\d\d(\d\d)?)" />
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- 4.3.4 -->
<xs:element name="value-date-and-or-time" abstract="true"/>
<!-- 4.3.5 -->
<xs:complexType name="value-timestamp">
    <xs:sequence>
        <xs:element ref="ns1:timestamp"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="timestamp">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:pattern value="\d{8}T\d{6}(Z|[\+-]\d\d(\d\d)?)" />
        </xs:restriction>
    </xs:simpleType>
</xs:element>
<!-- 4.4 -->
<xs:element name="boolean" type="xs:boolean"/>
<!-- 4.5 -->
<xs:element name="integer" type="xs:integer"/>
<!-- 4.6 -->
<xs:element name="float" type="xs:float"/>
<!-- 4.7 -->
<xs:element name="utc-offset">

```

```

    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern value="[+\-]\d\d(\d\d)?"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <!-- 4.8 -->
  <xs:element name="language-tag">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern
value="([a-z]{2,3}((-[a-z]{3}){0,3})?|[a-z]{4,8})
(-[a-z]{4})?((-[a-z]{2}|\d{3}))?(-([0-9a-z]{5,8}|
\d[0-9a-z]{3}))*(-([0-9a-wyz](-[0-9a-z]{2,8})+)*
(-x(-[0-9a-z]{1,8})+)?|x(-[0-9a-z]{1,8})+|[a-z]{1,3}
(-[0-9a-z]{2,8}){1,2})"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <!--

    5.1
  -->
  <xs:group name="param-language">
    <xs:annotation>
      <xs:documentation>Section 5: Parameters</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element minOccurs="0" ref="ns1:language"/>
    </xs:sequence>
  </xs:group>
  <xs:element name="language">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="ns1:language-tag"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <!-- 5.2 -->
  <xs:group name="param-pref">
    <xs:sequence>
      <xs:element minOccurs="0" ref="ns1:pref"/>
    </xs:sequence>
  </xs:group>
  <xs:element name="pref">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="integer">

```

```
        <xs:simpleType>
          <xs:restriction base="xs:integer">
            <xs:minInclusive value="1"/>
            <xs:maxInclusive value="100"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 5.4 -->
<xs:group name="param-altid">
  <xs:sequence>
    <xs:element minOccurs="0" ref="ns1:altid"/>
  </xs:sequence>
</xs:group>
<xs:element name="altid">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="ns1:text"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 5.5 -->
<xs:group name="param-pid">
  <xs:sequence>
    <xs:element minOccurs="0" ref="ns1:pid"/>
  </xs:sequence>
</xs:group>
<xs:element name="pid">
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="unbounded" name="text">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:pattern value="\d+(\.\d+)?"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 5.6 -->
<xs:group name="param-type">
  <xs:sequence>
    <xs:element minOccurs="0" ref="ns1:type"/>
  </xs:sequence>
</xs:group>
```

```
<xs:element name="type">
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="unbounded" name="text">
        <xs:simpleType>
          <xs:restriction base="xs:token">
            <xs:enumeration value="work"/>
            <xs:enumeration value="home"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 5.7 -->
<xs:group name="param-mediatype">
  <xs:sequence>
    <xs:element minOccurs="0" ref="ns1:mediatype"/>
  </xs:sequence>
</xs:group>
<xs:element name="mediatype">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="ns1:text"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 5.8 -->
<xs:group name="param-calscale">
  <xs:sequence>
    <xs:element minOccurs="0" ref="ns1:calscale"/>
  </xs:sequence>
</xs:group>
<xs:element name="calscale">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="text">
        <xs:simpleType>
          <xs:restriction base="xs:token">
            <xs:enumeration value="gregorian"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 5.9 -->
<xs:group name="param-sort-as">
```

```
<xs:sequence>
  <xs:element minOccurs="0" ref="ns1:sort-as"/>
</xs:sequence>
</xs:group>
<xs:element name="sort-as">
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="unbounded" ref="ns1:text"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 5.10 -->
<xs:group name="param-geo">
  <xs:sequence>
    <xs:element minOccurs="0" name="geo">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="ns1:uri"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:group>
<!-- 5.11 -->
<xs:group name="param-tz">
  <xs:sequence>
    <xs:element minOccurs="0" name="tz">
      <xs:complexType>
        <xs:choice>
          <xs:element ref="ns1:text"/>
          <xs:element ref="ns1:uri"/>
        </xs:choice>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:group>
<!--

6.1.3
-->
<xs:element name="source">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
        <xs:group ref="ns1:param-pref"/>
        <xs:group ref="ns1:param-mediatype"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element ref="ns1:uri"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- 6.1.4 -->
<xs:element name="kind">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="text">
        <xs:simpleType>
          <xs:union memberTypes="ns1:x-name ns1:iana-token">
            <xs:simpleType>
              <xs:restriction base="xs:token">
                <xs:enumeration value="individual"/>
              </xs:restriction>
            </xs:simpleType>
            <xs:simpleType>
              <xs:restriction base="xs:token">
                <xs:enumeration value="group"/>
              </xs:restriction>
            </xs:simpleType>
            <xs:simpleType>
              <xs:restriction base="xs:token">
                <xs:enumeration value="org"/>
              </xs:restriction>
            </xs:simpleType>
            <xs:simpleType>
              <xs:restriction base="xs:token">
                <xs:enumeration value="location"/>
              </xs:restriction>
            </xs:simpleType>
          </xs:union>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.2.1 -->
<xs:element name="fn">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
```

```
        <xs:sequence>
          <xs:group ref="ns1:param-language"/>
          <xs:group ref="ns1:param-altid"/>
          <xs:group ref="ns1:param-pid"/>
          <xs:group ref="ns1:param-pref"/>
          <xs:group ref="ns1:param-type"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element ref="ns1:text"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<!-- 6.2.2 -->
<xs:element name="n">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-language"/>
            <xs:group ref="ns1:param-sort-as"/>
            <xs:group ref="ns1:param-altid"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element maxOccurs="unbounded" ref="ns1:surname"/>
      <xs:element maxOccurs="unbounded" ref="ns1:given"/>
      <xs:element maxOccurs="unbounded" ref="ns1:additional"/>
      <xs:element maxOccurs="unbounded" ref="ns1:prefix"/>
      <xs:element maxOccurs="unbounded" ref="ns1:suffix"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="surname" type="xs:string"/>
<xs:element name="given" type="xs:string"/>
<xs:element name="additional" type="xs:string"/>
<xs:element name="prefix" type="xs:string"/>
<xs:element name="suffix" type="xs:string"/>
<!-- 6.2.3 -->
<xs:element name="nickname">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-language"/>
            <xs:group ref="ns1:param-altid"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```



```
        <xs:group ref="ns1:param-pid"/>
        <xs:group ref="ns1:param-pref"/>
        <xs:group ref="ns1:param-type"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:group ref="ns1:value-text-list"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- 6.2.4 -->
<xs:element name="photo">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:group ref="ns1:param-type"/>
            <xs:group ref="ns1:param-mediatype"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element ref="ns1:uri"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.2.5 -->
<xs:element name="bday">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-calscale"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:choice>
        <xs:element ref="ns1:value-date-and-or-time"/>
        <xs:element ref="ns1:text"/>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
<!-- 6.2.6 -->
<xs:element name="anniversary">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-calscale"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:choice>
        <xs:element ref="ns1:value-date-and-or-time"/>
        <xs:element ref="ns1:text"/>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.2.7 -->
<xs:element name="gender">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="ns1:sex"/>
      <xs:element minOccurs="0" ref="ns1:identity"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="sex">
  <xs:simpleType>
    <xs:restriction base="xs:token">
      <xs:enumeration value=""/>
      <xs:enumeration value="M"/>
      <xs:enumeration value="F"/>
      <xs:enumeration value="O"/>
      <xs:enumeration value="N"/>
      <xs:enumeration value="U"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="identity" type="xs:string"/>
<!-- 6.3.1 -->
<xs:group name="param-label">
  <xs:sequence>
    <xs:element minOccurs="0" ref="ns1:label"/>
  </xs:sequence>
</xs:group>
<xs:element name="label">
```

```
<xs:complexType>
  <xs:sequence>
    <xs:element ref="ns1:text"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="adr">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-language"/>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:group ref="ns1:param-type"/>
            <xs:group ref="ns1:param-geo"/>
            <xs:group ref="ns1:param-tz"/>
            <xs:group ref="ns1:param-label"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element maxOccurs="unbounded" ref="ns1:pobox"/>
      <xs:element maxOccurs="unbounded" ref="ns1:ext"/>
      <xs:element maxOccurs="unbounded" ref="ns1:street"/>
      <xs:element maxOccurs="unbounded" ref="ns1:locality"/>
      <xs:element maxOccurs="unbounded" ref="ns1:region"/>
      <xs:element maxOccurs="unbounded" ref="ns1:code"/>
      <xs:element maxOccurs="unbounded" ref="ns1:country"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="pobox" type="xs:string"/>
<xs:element name="ext" type="xs:string"/>
<xs:element name="street" type="xs:string"/>
<xs:element name="locality" type="xs:string"/>
<xs:element name="region" type="xs:string"/>
<xs:element name="code" type="xs:string"/>
<xs:element name="country" type="xs:string"/>
<!-- 6.4.1 -->
<xs:element name="tel">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>

```

```
<xs:group ref="ns1:param-pid"/>
<xs:group ref="ns1:param-pref"/>
<xs:element minOccurs="0" name="type">
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="unbounded" name="text"
        type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:group ref="ns1:param-mediatype"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:choice>
  <xs:element ref="ns1:text"/>
  <xs:element ref="ns1:uri"/>
</xs:choice>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- 6.4.2 -->
<xs:element name="email">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:group ref="ns1:param-type"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element ref="ns1:text"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.4.3 -->
<xs:element name="impp">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
```

```
        <xs:group ref="ns1:param-pref"/>
        <xs:group ref="ns1:param-type"/>
        <xs:group ref="ns1:param-mediatype"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
    <xs:element ref="ns1:uri"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- 6.4.4 -->
<xs:element name="lang">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="parameters">
                <xs:complexType>
                    <xs:sequence>
                        <xs:group ref="ns1:param-altid"/>
                        <xs:group ref="ns1:param-pid"/>
                        <xs:group ref="ns1:param-pref"/>
                        <xs:group ref="ns1:param-type"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:element ref="ns1:language-tag"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<!-- 6.5.1 -->
<xs:group name="property-tz">
    <xs:sequence>
        <xs:element name="tz">
            <xs:complexType>
                <xs:sequence>
                    <xs:element minOccurs="0" name="parameters">
                        <xs:complexType>
                            <xs:sequence>
                                <xs:group ref="ns1:param-altid"/>
                                <xs:group ref="ns1:param-pid"/>
                                <xs:group ref="ns1:param-pref"/>
                                <xs:group ref="ns1:param-type"/>
                                <xs:group ref="ns1:param-mediatype"/>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                    <xs:choice>
                        <xs:element ref="ns1:text"/>
                        <xs:element ref="ns1:uri"/>
                    </xs:choice>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:group>
```

```
        <xs:element ref="ns1:utc-offset"/>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:group>
<!-- 6.5.2 -->
<xs:group name="property-geo">
  <xs:sequence>
    <xs:element name="geo">
      <xs:complexType>
        <xs:sequence>
          <xs:element minOccurs="0" name="parameters">
            <xs:complexType>
              <xs:sequence>
                <xs:group ref="ns1:param-altid"/>
                <xs:group ref="ns1:param-pid"/>
                <xs:group ref="ns1:param-pref"/>
                <xs:group ref="ns1:param-type"/>
                <xs:group ref="ns1:param-mediatype"/>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
          <xs:element ref="ns1:uri"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:group>
<!-- 6.6.1 -->
<xs:element name="title">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-language"/>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:group ref="ns1:param-type"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element ref="ns1:text"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:group>
</xs:complexType>
```

```
</xs:element>
<!-- 6.6.2 -->
<xs:element name="role">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-language"/>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:group ref="ns1:param-type"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element ref="ns1:text"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.6.3 -->
<xs:element name="logo">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-language"/>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:group ref="ns1:param-type"/>
            <xs:group ref="ns1:param-mediatype"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element ref="ns1:uri"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.6.4 -->
<xs:element name="org">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-language"/>
```

```
        <xs:group ref="ns1:param-altid"/>
        <xs:group ref="ns1:param-pid"/>
        <xs:group ref="ns1:param-pref"/>
        <xs:group ref="ns1:param-type"/>
        <xs:group ref="ns1:param-sort-as"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:group ref="ns1:value-text-list"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- 6.6.5 -->
<xs:element name="member">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:group ref="ns1:param-mediatype"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element ref="ns1:uri"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.6.6 -->
<xs:element name="related">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:element minOccurs="0" name="type">
              <xs:complexType>
                <xs:sequence>
                  <xs:element maxOccurs="unbounded" name="text">
                    <xs:simpleType>
                      <xs:restriction base="xs:token">
                        <xs:enumeration value="work"/>
                        <xs:enumeration value="home"/>

```



```

        <xs:enumeration value="contact"/>
        <xs:enumeration value="acquaintance"/>
        <xs:enumeration value="friend"/>
        <xs:enumeration value="met"/>
        <xs:enumeration value="co-worker"/>
        <xs:enumeration value="colleague"/>
        <xs:enumeration value="co-resident"/>
        <xs:enumeration value="neighbor"/>
        <xs:enumeration value="child"/>
        <xs:enumeration value="parent"/>
        <xs:enumeration value="sibling"/>
        <xs:enumeration value="spouse"/>
        <xs:enumeration value="kin"/>
        <xs:enumeration value="muse"/>
        <xs:enumeration value="crush"/>
        <xs:enumeration value="date"/>
        <xs:enumeration value="sweetheart"/>
        <xs:enumeration value="me"/>
        <xs:enumeration value="agent"/>
        <xs:enumeration value="emergency"/>
    </xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:group ref="ns1:param-mediatype"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:choice>
    <xs:element ref="ns1:uri"/>
    <xs:element ref="ns1:text"/>
</xs:choice>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- 6.7.1 -->
<xs:element name="categories">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" name="parameters">
                <xs:complexType>
                    <xs:sequence>
                        <xs:group ref="ns1:param-altid"/>
                        <xs:group ref="ns1:param-pid"/>
                        <xs:group ref="ns1:param-pref"/>
                        <xs:group ref="ns1:param-type"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>
</xs:element>
```

```
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:group ref="ns1:value-text-list"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<!-- 6.7.2 -->
<xs:element name="note">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-language"/>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:group ref="ns1:param-type"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element ref="ns1:text"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.7.3 -->
<xs:element name="prodid">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="ns1:text"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.7.4 -->
<xs:element name="rev" type="ns1:value-timestamp"/>
<!-- 6.7.5 -->
<xs:element name="sound">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-language"/>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:group ref="ns1:param-type"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
        <xs:group ref="ns1:param-mediatype"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element ref="ns1:uri"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- 6.7.6 -->
<xs:element name="uid">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="ns1:uri"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.7.7 -->
<xs:element name="clientpidmap">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="ns1:sourceid"/>
      <xs:element ref="ns1:uri"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="sourceid" type="xs:positiveInteger"/>
<!-- 6.7.8 -->
<xs:element name="url">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:group ref="ns1:param-type"/>
            <xs:group ref="ns1:param-mediatype"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element ref="ns1:uri"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.8.1 -->
<xs:element name="key">
  <xs:complexType>
```

```
<xs:sequence>
  <xs:element minOccurs="0" name="parameters">
    <xs:complexType>
      <xs:sequence>
        <xs:group ref="ns1:param-altid"/>
        <xs:group ref="ns1:param-pid"/>
        <xs:group ref="ns1:param-pref"/>
        <xs:group ref="ns1:param-type"/>
        <xs:group ref="ns1:param-mediatype"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:choice>
    <xs:element ref="ns1:uri"/>
    <xs:element ref="ns1:text"/>
  </xs:choice>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- 6.9.1 -->
<xs:element name="fburl">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
            <xs:group ref="ns1:param-type"/>
            <xs:group ref="ns1:param-mediatype"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element ref="ns1:uri"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- 6.9.2 -->
<xs:element name="caladruri">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="ns1:param-altid"/>
            <xs:group ref="ns1:param-pid"/>
            <xs:group ref="ns1:param-pref"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
        <xs:group ref="nsl:param-type"/>
        <xs:group ref="nsl:param-mediatype"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element ref="nsl:uri"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<!-- 6.9.3 -->
<xs:element name="caluri">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="parameters">
        <xs:complexType>
          <xs:sequence>
            <xs:group ref="nsl:param-altid"/>
            <xs:group ref="nsl:param-pid"/>
            <xs:group ref="nsl:param-pref"/>
            <xs:group ref="nsl:param-type"/>
            <xs:group ref="nsl:param-mediatype"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element ref="nsl:uri"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- Top-level grammar -->
<xs:group name="property">
  <xs:choice>
    <xs:element ref="nsl:adr"/>
    <xs:element ref="nsl:anniversary"/>
    <xs:element ref="nsl:bday"/>
    <xs:element ref="nsl:caladruri"/>
    <xs:element ref="nsl:caluri"/>
    <xs:element ref="nsl:categories"/>
    <xs:element ref="nsl:clientpidmap"/>
    <xs:element ref="nsl:email"/>
    <xs:element ref="nsl:fburl"/>
    <xs:element ref="nsl:fn"/>
    <xs:group ref="nsl:property-geo"/>
    <xs:element ref="nsl:impp"/>
    <xs:element ref="nsl:key"/>
    <xs:element ref="nsl:kind"/>
    <xs:element ref="nsl:lang"/>
    <xs:element ref="nsl:logo"/>
    <xs:element ref="nsl:member"/>
```

```
<xs:element ref="ns1:n"/>
<xs:element ref="ns1:nickname"/>
<xs:element ref="ns1:note"/>
<xs:element ref="ns1:org"/>
<xs:element ref="ns1:photo"/>
<xs:element ref="ns1:prodid"/>
<xs:element ref="ns1:related"/>
<xs:element ref="ns1:rev"/>
<xs:element ref="ns1:role"/>
<xs:element ref="ns1:gender"/>
<xs:element ref="ns1:sound"/>
<xs:element ref="ns1:source"/>
<xs:element ref="ns1:tel"/>
<xs:element ref="ns1:title"/>
<xs:group ref="ns1:property-tz"/>
<xs:element ref="ns1:uid"/>
<xs:element ref="ns1:url"/>
</xs:choice>
</xs:group>

<xs:element name="vcards">
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="unbounded" ref="ns1:vcard"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

<xs:complexType name="vcardType">
  <xs:complexContent>
    <xs:restriction base="xs:anyType">
      <xs:choice maxOccurs="unbounded">
        <xs:group ref="ns1:property"/>
        <xs:element ref="ns1:group"/>
      </xs:choice>
    </xs:restriction>
  </xs:complexContent>
</xs:complexType>

<xs:element name="vcard" type="ns1:vcardType"/>

<xs:element name="group">
  <xs:complexType>
    <xs:group minOccurs="0" maxOccurs="unbounded"
      ref="ns1:property"/>
    <xs:attribute name="name" use="required"/>
  </xs:complexType>
```

```
</xs:element>  
</xs:schema>
```

## Appendix B. XML Validation

This document defines a number of XML schemas and contains various examples. Extracting the XML and validating the examples against the schemas can be challenging, especially due to the formatting limitations introduced by IETF RFCs. For those readers who copy the XML schemas and examples directly from this document, please consider that errors might be introduced due to line breaks and extra whitespaces in the regular expressions contained in the vcard schema in Appendix A. To validate the PIDF-LO from Figure 18 it is also necessary to consult the referenced RFCs and copy the schemas necessary for successful validation.

The XML schemas found in this document include a 'SchemaLocation' attribute. Depending on the location of the downloaded schema files you may need to adjust this schema location or configure your XML editor to point to the location.

For convenience of readers, the schemas are available at <http://ip-emergency.net/additional-data.zip> and the XML examples are available at the IETF ECRIT Working Group wiki page [ECRIT-WG-wiki].

Note to RFC Editor: After IANA has published the schemas, the above link to the schemas should be replaced with [IANA-XML-Schemas].

## Authors' Addresses

Randall Gellens  
San Diego, CA 92121  
US

Email: [rg+ietf@randy.pensive.org](mailto:rg+ietf@randy.pensive.org)

Brian Rosen  
NeuStar  
470 Conrad Dr.  
Mars, PA 16046  
US

Phone: +1 724 382 1051  
Email: [br@brianrosen.net](mailto:br@brianrosen.net)

Hannes Tschofenig  
Hall in Tirol 6060  
Austria

Email: Hannes.tschofenig@gmx.net  
URI: <http://www.tschofenig.priv.at>

Roger Marshall  
TeleCommunication Systems, Inc.  
2401 Elliott Avenue  
Seattle, WA 98121  
US

Phone: +1 206 792 2424  
Email: [rmarshall@telecomsys.com](mailto:rmarshall@telecomsys.com)  
URI: <http://www.telecomsys.com>

James Winterbottom  
AU

Email: [a.james.winterbottom@gmail.com](mailto:a.james.winterbottom@gmail.com)



ECRIT  
Internet-Draft  
Intended status: Standards Track  
Expires: September 10, 2020

B. Rosen  
  
H. Schulzrinne  
Columbia U.  
H. Tschofenig  
ARM Limited  
R. Gellens  
Core Technology Consulting  
March 9, 2020

Non-Interactive Emergency Calls  
draft-ietf-ecrit-data-only-ea-22

Abstract

Use of the Internet for emergency calling is described in RFC 6443, 'Framework for Emergency Calling Using Internet Multimedia'. In some cases of emergency calls, the transmission of application data is all that is needed and no interactive media channel is established: a situation referred to as 'non-interactive emergency calls', where, unlike most emergency calls, there is no two way interactive media such as voice or video or text. This document describes use of a SIP MESSAGE transaction that includes a container for the data based on the Common Alerting Protocol (CAP). That type of emergency request does not establish a session, distinguishing it from SIP INVITE, which does. Any device that needs to initiate a request for emergency services without an interactive media channel would use the mechanisms in this document.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 10, 2020.

## Copyright Notice

Copyright (c) 2020 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	3
3. Architectural Overview . . . . .	4
4. Protocol Specification . . . . .	6
4.1. CAP Transport . . . . .	6
4.2. Profiling of the CAP Document Content . . . . .	7
4.3. Sending a non-interactive Emergency Call . . . . .	8
5. Error Handling . . . . .	9
5.1. 425 (Bad Alert Message) Response Code . . . . .	9
5.2. The AlertMsg-Error Header Field . . . . .	9
6. Call Backs . . . . .	11
7. Handling Large Amounts of Data . . . . .	11
8. Example . . . . .	12
9. Security Considerations . . . . .	16
10. IANA Considerations . . . . .	18
10.1. Registration of the 'application/EmergencyCallData.cap+xml' media type . . . .	18
10.2. IANA Registration of 'cap' Additional Data Block . . . .	19
10.3. IANA Registration for 425 Response Code . . . . .	19
10.4. IANA Registration of New AlertMsg-Error Header Field . .	20
10.5. IANA Registration for the SIP AlertMsg-Error Codes . . .	20
11. Acknowledgments . . . . .	21
12. References . . . . .	21
12.1. Normative References . . . . .	21
12.2. Informative References . . . . .	23
Authors' Addresses . . . . .	23

## 1. Introduction

[RFC6443] describes how devices use the Internet to place emergency calls and how Public Safety Answering Points (PSAPs) handle Internet multimedia emergency calls natively. The exchange of multimedia traffic for emergency services involves a SIP session establishment starting with a SIP INVITE that negotiates various parameters for that session.

In some cases, however, there is only application data to be conveyed from the end devices to a PSAP or an intermediary. Examples of such environments include sensors issuing alerts, and certain types of medical monitors. These messages may be one-shot alerts to emergency authorities and do not require establishment of a session. These types of interactions are called 'non-interactive emergency calls'. In this document, we use the term "call" so that similarities between non-interactive alerts and sessions with interactive media are more obvious.

Non-interactive emergency calls are similar to regular emergency calls in the sense that they require the emergency indications, emergency call routing functionality and location. However, the communication interaction will not lead to the exchange of interactive media, that is, Real-Time Protocol packets, such as voice, video data or real-time text.

The Common Alerting Protocol (CAP) [cap] is a format for exchanging emergency alerts and public warnings. CAP is mainly used for conveying alerts and warnings between authorities and from authorities to citizens/individuals. This document is concerned with citizen-to-authority "alerts", where the alert is a call without any interactive media.

This document describes a method of including a CAP message in a SIP transaction by defining it as a block of "additional data" as defined in [RFC7852]. The CAP message is included either by value (the CAP message is in the body of the message, using a CID) or by reference (the message includes a URI that, when dereferenced, returns the CAP message). The additional data mechanism is also used to send alert-specific data beyond that available in the CAP message. This document also describes how a SIP MESSAGE [RFC3428] transaction can be used to send a non-interactive call.

## 2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP

14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

A non-interactive emergency call is an emergency call where there is no two-way interactive media.

SIP is the Session Initiation Protocol [RFC3261]

PIDF-LO is Presence Information Data Format - Location Object, a data structure for carrying location [RFC4119]

LoST is the Location To Service Translation protocol [RFC5222]

CID is Content-ID [RFC2392]

CAP is the Common Alerting Protocol [cap]

PSAP is a Public Safety Answering Point, the call center for emergency calls.

ESRP is an Emergency Services Routing Proxy, a type of SIP Proxy Server used in some emergency services networks

### 3. Architectural Overview

This section illustrates two envisioned usage modes: targeted and location-based emergency alert routing.

1. Emergency alerts containing only data are targeted to an intermediary recipient responsible for evaluating the next steps. These steps could include:
  1. Sending a non-interactive call containing only data towards a Public Safety Answering Point (PSAP);
  2. Establishing a third-party-initiated emergency call towards a PSAP that could include audio, video, and data.
2. Emergency alerts may be targeted to a Service URN [RFC5031] used for IP-based emergency calls where the recipient is not known to the originator. In this scenario, the alert may contain only data (e.g., a CAP, Geolocation header field and one or more Call-Info header fields containing Additional Data [RFC7852] in a SIP MESSAGE).

Figure 1 shows a deployment variant where a sensor is pre-configured (using techniques outside the scope of this document) to issue an alert to an aggregator that processes these messages and performs

whatever steps are necessary to appropriately react to the alert. For example, a security firm may use different sensor inputs to dispatch their security staff to a building they protect or to initiate a third-party emergency call.

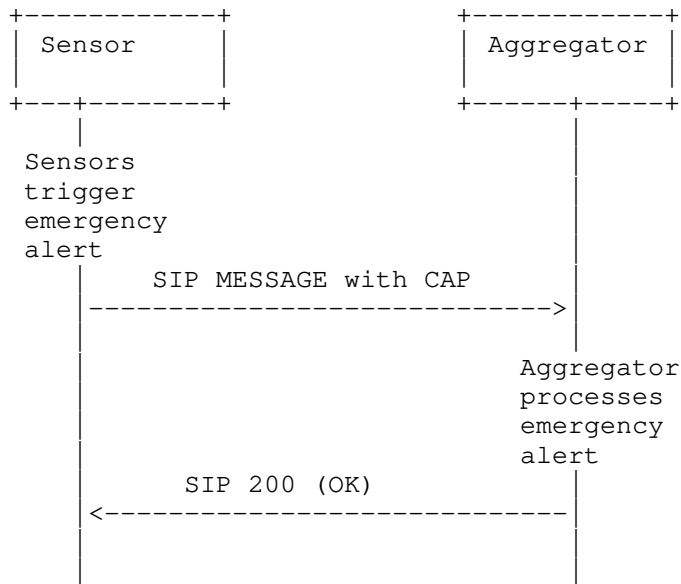


Figure 1: Targeted Emergency Alert Routing

In Figure 2 a scenario is shown whereby the alert is routed using location information and a Service URN. An emergency services routing proxy (ESRP) may use LoST (a protocol defined by [RFC5222] which translates a location to a URI used to route an emergency call) to determine the next-hop proxy to route the alert message to. A possible receiver is a PSAP and the recipient of the alert may be a call taker. In the generic case, there is very likely no prior relationship between the originator and the receiver, e.g., a PSAP. For example, a PSAP is likely to receive and accept alerts from entities it has no previous relationship with. This scenario is similar to a classic voice emergency services call and the description in [RFC6881] is applicable. In this use case, the only difference between an emergency call and an emergency non-interactive call is that the former uses INVITE, creates a session, and negotiates one or more media streams, while the latter uses MESSAGE, does not create a session, and does not have interactive media.

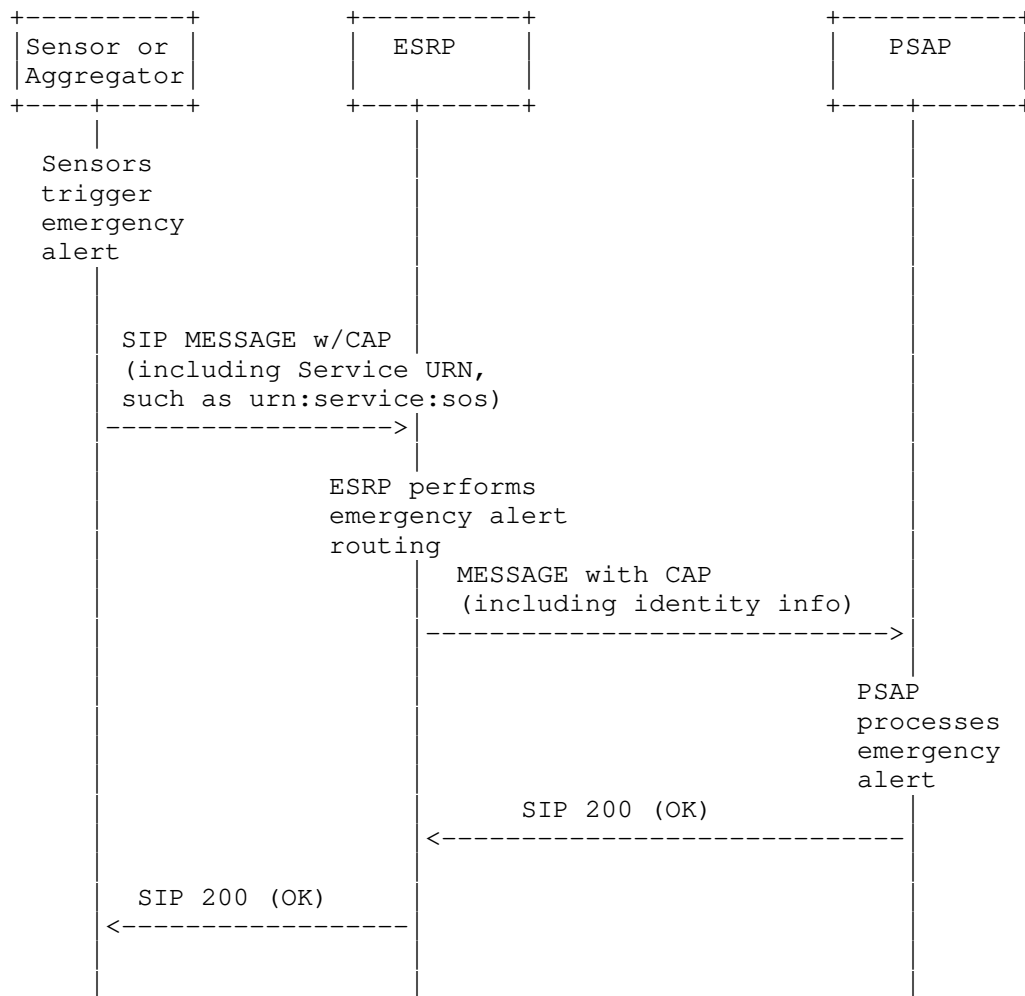


Figure 2: Location-Based Emergency Alert Routing

## 4. Protocol Specification

### 4.1. CAP Transport

A CAP message is sent in the initial message of any SIP transaction. However, this document only addresses sending a CAP message in a SIP MESSAGE transaction for a one-shot, non-interactive emergency call. Behavior with other transactions is not defined.

The CAP message is included in a SIP message as an additional-data block [RFC7852]. Accordingly, it is introduced to the SIP message

with a Call-Info header field with a purpose of "EmergencyCallData.cap". The header field may contain a URI that is used by the recipient (or in some cases, an intermediary) to obtain the CAP message. Alternatively, the Call-Info header field may contain a Content-ID url [RFC2392] and the CAP message included in the body of the message. In the latter case, the CAP message is located in a MIME block of the type 'application/emergencyCallData.cap+xml'.

If the SIP server does not support the functionality required to fulfill the request then a 501 Not Implemented will be returned as specified in [RFC3261]. This is the appropriate response when a User Agent Server (UAS) does not recognize the request method and is not capable of supporting it for any user.

The 415 Unsupported Media Type error will be returned as specified in [RFC3261] if the SIP server is refusing to service the request because the message body of the request is in a format not supported by the server for the requested method. The server MUST return a list of acceptable formats using the Accept, Accept-Encoding, or Accept-Language header fields, depending on the specific problem with the content.

#### 4.2. Profiling of the CAP Document Content

The usage of CAP MUST conform to the specification provided with [cap]. For usage with SIP the following additional requirements are imposed (where "sender" and "author" are as defined in CAP and "Originator" is the entity sending the alert):

sender: The following restrictions and conditions apply to setting the value of the <sender> element:

- \* Originator is a SIP entity, Author indication irrelevant: When the alert was created by a SIP-based originator and it is not useful to be explicit about the author of the alert, then the <sender> element MUST be populated with the SIP URI of the user agent.
- \* Originator is a non-SIP entity, Author indication irrelevant: When the alert was created by a non-SIP based entity and the identity of this original sender is to be preserved, then this identity MUST be placed into the <sender> element. In this situation it is not useful to be explicit about the author of the alert. The specific type of identity being used will depend on the technology used by the original originator.

- \* Author indication relevant: When the author is different from the actual originator of the message and this distinction should be preserved, then the <sender> element MUST NOT contain the SIP URI of the user agent.

incidents: The <incidents> element MUST be present. This incident identifier MUST be chosen in such a way that it is unique for a given <sender, expires, incidents> combination. Note that the <expires> element is OPTIONAL and might not be present.

scope: The value of the <scope> element MAY be set to "Private" if the alert is not meant for public consumption. The <addresses> element is, however, not used by this specification since the message routing is performed by SIP and the respective address information is already available in other SIP header fields. Populating information twice into different parts of the message may lead to inconsistency.

parameter: The <parameter> element MAY contain additional information specific to the sender, conforming to the CAP message syntax.

area: It is RECOMMENDED to omit this element when constructing a message. If the CAP message is given to the SIP entity to transport and it already contains an <area> element, then the specified location information SHOULD be copied into a PIDF-LO structure (the data format for location used by emergency calls on the Internet) referenced by the SIP 'Geolocation' header field. If the CAP message is being created by the SIP entity using a PIDF-LO structure referenced by 'geolocation' to construct <area>, implementers must be aware that <area> is limited to a circle or polygon, and conversion of other shapes will be required. Points SHOULD be converted to a circle with a radius equal to the uncertainty of the point. Arc- bands and ellipses SHOULD be converted to polygons with similar coverage, and 3D locations SHOULD be converted to 2D forms with similar coverage.

#### 4.3. Sending a non-interactive Emergency Call

A non-interactive emergency call is sent using a SIP MESSAGE transaction with a CAP URI or body part as described above in a manner similar to how an emergency call with interactive media is sent, as described in [RFC6881]. The MESSAGE transaction does not create a session nor establish interactive media streams, but



otherwise, the header content of the transaction, routing, and processing of non-interactive calls are the same as those of other emergency calls.

## 5. Error Handling

This section defines a new error response code and a header field for additional information.

### 5.1. 425 (Bad Alert Message) Response Code

This SIP extension creates a new location-specific response code, defined as follows:

#### 425 (Bad Alert Message)

The 425 response code is a rejection of the request, indicating that it was malformed enough that no reasonable emergency response to the alert can be determined.

A SIP intermediary can also this code to reject an alert it receives from a User Agent (UA) when it detects that the provided alert is malformed.

Section 5.2 describes an AlertMsg-Error header field with more details about what was wrong with the alert message in the request. This header field **MUST** be included in the 425 response.

It is usually the case that emergency calls are not rejected if there is any useful information that can be acted upon. It is only appropriate to generate a 425 response when the responding entity has no other information in the request that is usable by the responder.

A 425 response code **MUST NOT** be sent in response to a request that lacks an alert message, as the user agent in that case may not support this extension.

A 425 response is a final response within a transaction, and **MUST NOT** terminate an existing dialog.

### 5.2. The AlertMsg-Error Header Field

The AlertMsg-Error header field provides additional information about what was wrong with the original request. In some cases the provided information will be used for debugging purposes.

The AlertMsg-Error header field has the following ABNF [RFC5234]:

```
message-header    =/ AlertMsg-Error
                   ; (message-header from RFC3261)
AlertMsg-Error    = "AlertMsg-Error" HCOLON
                   ErrorValue
ErrorValue        = error-code
                   *(SEMI error-params)
error-code        = 3DIGIT
error-params      = error-code-text
                   / generic-param ; from RFC3261
error-code-text   = "message" EQUAL quoted-string ; from RFC3261
```

HCOLON, SEMI, and EQUAL are defined in [RFC3261]. DIGIT is defined in [RFC5234].

The AlertMsg-Error header field MUST contain only one ErrorValue to indicate what was wrong with the alert payload the recipient determined was bad.

The ErrorValue contains a 3-digit error code indicating what was wrong with the alert in the request. This error code has a corresponding quoted error text string that is human readable. The text string is OPTIONAL, but RECOMMENDED for human readability, similar to the string phrase used for SIP response codes. The strings in this document are recommendations, and are not standardized -- meaning an operator can change the strings -- but MUST NOT change the meaning of the error code. The code space for ErrorValue is separate from SIP Status Codes.

The AlertMsg-Error header field MAY be included in any response if an alert message was in the request part of the same transaction. For example, suppose a UA includes an alert in a MESSAGE to a PSAP. The PSAP can accept this MESSAGE, even though its UA determined that the alert message contained in the MESSAGE was bad. The PSAP merely includes an AlertMsg-Error header field value in the 200 OK to the MESSAGE, thus informing the UA that the MESSAGE was accepted but the alert provided was bad.

If, on the other hand, the PSAP cannot accept the transaction without a suitable alert message, a 425 response is sent.

A SIP intermediary that requires the UA's alert message in order to properly process the transaction may also send a 425 with an AlertMsg-Error code.

This document defines an initial list of AlertMsg-Error values for any SIP response, including provisional responses (other than 100 Trying) and the new 425 response. There MUST NOT be more than one AlertMsg-Error code in a SIP response. AlertMsg-Error values sent in

provisional responses MUST be sent using the mechanism defined in [RFC3262]; or, if that mechanism is not negotiated, MUST be repeated in the final response to the transaction.

AlertMsg-Error: 100 ; message="Cannot Process the Alert Payload"

AlertMsg-Error: 101 ; message="Alert Payload was not present or could not be found"

AlertMsg-Error: 102 ; message="Not enough information to determine the purpose of the alert"

AlertMsg-Error: 103 ; message="Alert Payload was corrupted"

Additionally, if an entity cannot or chooses not to process the alert message from a SIP request, a 500 (Server Internal Error) SHOULD be used with or without a configurable Retry-After header field.

## 6. Call Backs

This document does not describe any method for the recipient to call back the sender of a non-interactive call. Usually, these alerts are sent by automata, which do not have a mechanism to receive calls of any kind. The identifier in the 'From' header field may be useful to obtain more information, but any such mechanism is not defined in this document. The CAP message may contain related contact information for the sender.

## 7. Handling Large Amounts of Data

It is not atypical for sensors to have large quantities of data that they may wish to send. Including large amounts of data (tens of kilobytes) in a MESSAGE is not advisable, because SIP entities are usually not equipped to handle very large messages. In such cases, the sender SHOULD make use of the by-reference mechanisms defined in [RFC7852], which involves making the data available via HTTPS [RFC2818] (either at the originator or at another entity), placing a URI to the data in the 'Call-Info' header field, and the recipient uses HTTPS to retrieve the data. The CAP message itself can be sent by reference using this mechanism, as can any or all of the Additional Data blocks that may contain sensor-specific data.

There are no rate limiting mechanisms for any SIP transactions that are standardized, although implementations often include such functions. Non-interactive emergency calls are typically handled the same as any emergency call, which means a human call-taker is involved. Implementations should take note of this limitation,

especially when calls are placed automatically without human initiation.

## 8. Example

The following example shows a CAP document indicating a BURGLARY alert issued by a sensor called 'sensor1@example.com'. The location of the sensor can be obtained from the attached location information provided via the 'geolocation' header field contained in the SIP MESSAGE structure. Additionally, the sensor provided some data along with the alert message, using proprietary information elements intended only to be processed by the receiver, a SIP entity acting as an aggregator.

```
MESSAGE sip:aggregator@example.com SIP/2.0
Via: SIP/2.0/TCP sensor1.example.com;branch=z9hG4bK776sgdkse
Max-Forwards: 70
From: sip:sensor1@example.com;tag=49583
To: sip:aggregator@example.com
Call-ID: asd88asd77a@2001:db8::ff
Geolocation: <cid:abcdef@example.com>
;routing-allowed=yes
Supported: geolocation
CSeq: 1 MESSAGE
Call-Info: cid:abcdef2@example.com;purpose=EmergencyCallData.cap
Content-Type: multipart/mixed; boundary=boundary1
Content-Length: ...

--boundary1
Content-Type: application/EmergencyCallData.cap+xml
Content-ID: <abcdef2@example.com>
Content-Disposition: by-reference;handling=optional

<?xml version="1.0" encoding="UTF-8"?>

<alert xmlns="urn:oasis:names:tc:emergency:cap:1.1">
  <identifier>S-1</identifier>
  <sender>sip:sensor1@example.com</sender>
  <sent>2020-01-04T20:57:35Z</sent>
  <status>Actual</status>
  <msgType>Alert</msgType>
  <scope>Private</scope>
  <incidents>abc1234</incidents>
  <info>
    <category>Security</category>
    <event>BURGLARY</event>
    <urgency>Expected</urgency>
```

```
<certainty>Likely</certainty>
<severity>Moderate</severity>
<senderName>SENSOR 1</senderName>
<parameter>
  <valueName>SENSOR-DATA-NAMESPACE1</valueName>
  <value>123</value>
</parameter>
<parameter>
  <valueName>SENSOR-DATA-NAMESPACE2</valueName>
  <value>TRUE</value>
</parameter>
</info>
</alert>

--boundary1
Content-Type: application/pidf+xml
Content-ID: <abcdef2@example.com>

<?xml version="1.0" encoding="UTF-8"?>
  <presence
    xmlns="urn:ietf:params:xml:ns:pidf"
    xmlns:gp="urn:ietf:params:xml:ns:pidf:geopriv10"
    xmlns:gbp=
      "urn:ietf:params:xml:ns:pidf:geopriv10:basicPolicy"
    xmlns:cl="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
    xmlns:gml="http://www.opengis.net/gml"
    xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
    entity="pres:alice@atlanta.example.com">
    <dm:device id="sensor">
      <gp:geopriv>
        <gp:location-info>
          <gml:location>
            <gml:Point srsName="urn:ogc:def:crs:EPSG::4326">
              <gml:pos>44.85249659 -93.238665712</gml:pos>
            </gml:Point>
          </gml:location>
        </gp:location-info>
        <gp:usage-rules>
          <gbp:retransmission-allowed>false
          </gbp:retransmission-allowed>
          <gbp:retention-expiry>2020-02-04T20:57:29Z
          </gbp:retention-expiry>
        </gp:usage-rules>
        <gp:method>802.11</gp:method>
      </gp:geopriv>
      <dm:timestamp>2020-01-04T20:57:29Z</dm:timestamp>
    </dm:device>
  </presence>
```

--boundary1--

Figure 3: Example Message conveying an Alert to an aggregator

The following shows the same CAP document sent as a non-interactive emergency call towards a PSAP.

```
MESSAGE urn:service:sos SIP/2.0
Via: SIP/2.0/TCP sip:aggreg.1.example.com;branch=z9hG4bK776abssa
Max-Forwards: 70
From: sip:aggregator@example.com;tag=32336
To: 112
Call-ID: asdf33443a@example.com
Route: sip:psap1.example.gov
Geolocation: <cid:abcdef@example.com>
;routing-allowed=yes
Supported: geolocation
Call-info: cid:abcdef2@example.com;purpose=EmergencyCallData.cap
CSeq: 1 MESSAGE
Content-Type: multipart/mixed; boundary=boundary1
Content-Length: ...
```

--boundary1

```
Content-Type: application/EmergencyCallData.cap+xml
Content-ID: <abcdef2@example.com>
<?xml version="1.0" encoding="UTF-8"?>

<alert xmlns="urn:oasis:names:tc:emergency:cap:1.1">
  <identifier>S-1</identifier>
  <sender>sip:sensor1@example.com</sender>
  <sent>2020-01-04T20:57:35Z</sent>
  <status>Actual</status>
  <msgType>Alert</msgType>
  <scope>Private</scope>
  <incidents>abc1234</incidents>
  <info>
    <category>Security</category>
    <event>BURGLARY</event>
    <urgency>Expected</urgency>
    <certainty>Likely</certainty>
    <severity>Moderate</severity>
    <senderName>SENSOR 1</senderName>
    <parameter>
      <valueName>SENSOR-DATA-NAMESPACE1</valueName>
      <value>123</value>
    </parameter>
```

```

    <parameter>
      <valueName>SENSOR-DATA-NAMESPACE2</valueName>
      <value>TRUE</value>
    </parameter>
  </info>
</alert>

--boundary1

Content-Type: application/pidf+xml
Content-ID: <abcdef2@example.com>
<?xml version="1.0" encoding="UTF-8"?>
  <presence
    xmlns="urn:ietf:params:xml:ns:pidf"
    xmlns:gp="urn:ietf:params:xml:ns:pidf:geopriv10"
    xmlns:gbp="urn:ietf:params:xml:ns:pidf:geopriv10:basicPolicy"
    xmlns:cl="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
    xmlns:gml="http://www.opengis.net/gml"
    xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
    entity="pres:alice@atlanta.example.com">
    <dm:device id="sensor">
      <gp:geopriv>
        <gp:location-info>
          <gml:location>
            <gml:Point srsName="urn:ogc:def:crs:EPSG::4326">
              <gml:pos>44.85249659 -93.2386657124</gml:pos>
            </gml:Point>
          </gml:location>
        </gp:location-info>
        <gp:usage-rules>
          <gbp:retransmission-allowed>false
        </gbp:retransmission-allowed>
          <gbp:retention-expiry>2020-02-04T20:57:25Z
        </gbp:retention-expiry>
        </gp:usage-rules>
        <gp:method>802.11</gp:method>
      </gp:geopriv>
      <dm:timestamp>2020-01-04T20:57:25Z</dm:timestamp>
    </dm:device>
  </presence>
--boundary1--

```

Figure 4: Example Message conveying an Alert to a PSAP

## 9. Security Considerations

This section discusses security considerations when SIP user agents issue emergency alerts utilizing MESSAGE and CAP. Location-specific threats are not unique to this document and are discussed in [RFC7378] and [RFC6442].

The ECRIT emergency services architecture [RFC6443] considers classic individual-to-authority emergency calling where the identity of the emergency caller does not play a role at the time of the call establishment itself, i.e., a response to the emergency call does not depend on the identity of the caller. In the case of emergency alerts generated by devices such as sensors, the processing may be different in order to reduce the number of falsely generated emergency alerts. Alerts could get triggered based on certain sensor input that might have been caused by factors other than the actual occurrence of an alert-relevant event. For example, a sensor may simply be malfunctioning. For this reason, not all alert messages are directly sent to a PSAP, but rather may be pre-processed by a separate entity, potentially under supervision by a human, to filter alerts and potentially correlate received alerts with others to obtain a larger picture of the ongoing situation.

In any case, for alerts initiated by sensors, the identity could play an important role in deciding whether to accept or ignore an incoming alert message. With the scenario shown in Figure 1 it is very likely that only authenticated sensor input will be processed. For this reason, it needs to be possible to refuse to accept alert messages from unknown origins. Two types of information elements can be used for this purpose:

1. SIP itself provides security mechanisms that allow the verification of the originator's identity, such as P-Asserted-Identity [RFC3325] or SIP Identity [RFC8224]. The latter provides a cryptographic assurance while the former relies on a chain of trust model. These mechanisms can be reused.
2. CAP provides additional security mechanisms and the ability to carry further information about the sender's identity. Section 3.3.4.1 of [cap] specifies the signing algorithms of CAP documents.

The specific policy and mechanisms used in a given deployment are out of scope for this document.

There is no rate limiting mechanisms in SIP, and all kinds of emergency calls, including those defined in this document could be used by malicious actors, or misbehaving devices to effect a denial



of service attack on the emergency services. The mechanism defined in this document does not introduce any new considerations although it may be more likely that devices that place non-interactive emergency calls without a human initiating them may be more likely than those that require a user to initiate them.

Implementors should note that automated emergency calls may be prohibited or regulated in some jurisdictions, and there may be penalties for "false positive" calls.

This document describes potential retrieval of information by dereferencing URIs found in a Call Info header of a SIP MESSAGE. These may include a CAP message as well as other Additional Data (RFC7852) blocks. The domain of the device sending the SIP MESSAGE, the domain of the server holding the CAP message, if sent by reference, and the domain of other Additional Data blocks, if sent by reference, may all be different. No assumptions can be made that there are trust relationships between these entities. Recipients MUST take precautions in retrieving any Additional Data blocks passed by reference, including the CAP message, because the URI may point to a malicious actor or entity not expecting to be referred to for this purpose. The considerations in handling URIs in [RFC3986] apply.

Use of timestamps to prevent replay is subject to the availability of accurate time at all participants. Because emergency event notification via this mechanism is relatively low frequency and generally involves human interaction, implementations may wish to consider messages with times within small number of seconds of each other to be effectively simultaneous for the purposes of detecting replay. Implementations may also wish to consider that most deployed time distribution protocols likely to be used by these systems are not presently secure.

In addition to the desire to perform identity-based access control, the classic communication security threats need to be considered, including integrity protection to prevent forgery or replay of alert messages in transit. To deal with replay of alerts, a CAP document contains the mandatory <identifier>, <sender>, <sent> elements and an optional <expire> element. Together, these elements make the CAP document unique for a specific sender and provide time restrictions. An entity that has already received a CAP message within the indicated timeframe is able to detect a replayed message and, if the content of that message is unchanged, then no additional security vulnerability is created. Additionally, it is RECOMMENDED to make use of SIP security mechanisms, such as the SIP Identity PASSport [RFC8225], to tie the CAP message to the SIP message. To provide protection of the entire SIP message exchange between neighboring SIP entities, the usage of TLS is RECOMMENDED. [RFC6443] discusses the

issues of using TLS with emergency calls, which are equally applicable to non-interactive emergency calls

Note that none of the security mechanisms in this document protect against a compromised sensor sending crafted alerts. Confidentiality provided for any emergency calls, including non-interactive messages, is subject to local regulations. Privacy issues are discussed in [RFC7852] and are applicable here.

## 10. IANA Considerations

### 10.1. Registration of the 'application/EmergencyCallData.cap+xml' media type

To: ietf-types@iana.org

Subject: Registration of media type application/  
EmergencyCallData.cap+xml

Type name: application

Subtype name: cap+xml

Required parameters: (none)

Optional parameters: charset; Indicates the character encoding of enclosed XML. Default is UTF-8 [RFC3629].

Encoding considerations: 7bit, 8bit or binary. See [RFC7303], Section 3.2.

Security considerations: This content type is designed to carry payloads of the Common Alerting Protocol (CAP). RFC XXX [Replace by the RFC number of this specification] discusses security considerations for this.

Interoperability considerations: This content type provides a way to convey CAP payloads.

Published specification: RFC XXX [Replace by the RFC number of this specification].

Applications which use this media type: Applications that convey alerts and warnings according to the CAP standard.

Fragment Identifier Considerations: N/A .

Additional information: OASIS has published the Common Alerting Protocol at [http://www.oasis-open.org/committees/documents.php&wg\\_abbrev=emergency](http://www.oasis-open.org/committees/documents.php&wg_abbrev=emergency)

Person and email address to contact for further information: Hannes Tschofenig, [hannes.tschofenig@gmx.net](mailto:hannes.tschofenig@gmx.net)

Intended usage: Limited use

Author/Change controller: The IESG

Other information: This media type is a specialization of application/xml [RFC7303], and many of the considerations described there also apply to application/cap+xml.

#### 10.2. IANA Registration of 'cap' Additional Data Block

This document registers a new block type in the sub-registry called 'Emergency Call Data Types' of the Emergency Call Additional Data Registry defined in [RFC7852]. The token is "cap", the Data About is "The Call" and the reference is this document.

#### 10.3. IANA Registration for 425 Response Code

In the SIP Response Codes registry, the following is added

Reference: RFC-XXXX (i.e., this document)

Response code: 425 (recommended number to assign)

Default reason phrase: Bad Alert Message

## Registry:

Response Code	Reference
-----	-----
Request Failure 4xx	
425 Bad Alert Message	[this doc]

This SIP Response code is defined in Section 5.

## 10.4. IANA Registration of New AlertMsg-Error Header Field

The SIP AlertMsg-error header field is created by this document, with its definition and rules in Section 5, to be added to the IANA Session Initiation Protocol (SIP) Parameters registry with two actions:

1. Update the Header Fields registry with

## Registry:

Header Name	compact	Reference
-----	-----	-----
AlertMsg-Error		[this doc]

2. In the portion titled "Header Field Parameters and Parameter Values", add

Header Field	Parameter Name	Predefined Values	Reference
-----	-----	-----	-----
AlertMsg-Error	code	no	[this doc]

## 10.5. IANA Registration for the SIP AlertMsg-Error Codes

This document creates a new registry for SIP, called "AlertMsg-Error Codes". AlertMsg-Error codes provide reasons for an error discovered by a recipient, categorized by the action to be taken by the error recipient. The initial values for this registry are shown below.

Registry Name: AlertMsg-Error Codes

Reference: [this doc]

Registration Procedures: Specification Required

Code	Default Reason Phrase	Reference
100	"Cannot Process the Alert Payload"	[this doc]
101	"Alert Payload was not present or could not be found"	[this doc]
102	"Not enough information to determine the purpose of the alert"	[this doc]
103	"Alert Payload was corrupted"	[this doc]

Details of these error codes are in Section 5.

## 11. Acknowledgments

The authors would like to thank the participants of the Early Warning adhoc meeting at IETF#69 for their feedback. Additionally, we would like to thank the members of the NENA Long Term Direction Working Group for their feedback.

Additionally, we would like to thank Martin Thomson, James Winterbottom, Shida Schubert, Bernard Aboba, Marc Linsner, Christer Holmberg and Ivo Sedlacek for their review comments.

## 12. References

### 12.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997.
- [cap] Jones, E. and A. Botterell, "Common Alerting Protocol v. 1.2", October 2005, <<https://docs.oasis-open.org/emergency/cap/v1.2/CAP-v1.2-os.pdf>>.
- [RFC2392] Levinson, E., "Content-ID and Message-ID Uniform Resource Locators", RFC 2392, DOI 10.17487/RFC2392, August 1998, <<https://www.rfc-editor.org/info/rfc2392>>.
- [RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, DOI 10.17487/RFC2818, May 2000, <<https://www.rfc-editor.org/info/rfc2818>>.
- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <<https://www.rfc-editor.org/info/rfc3261>>.

- [RFC3262] Rosenberg, J. and H. Schulzrinne, "Reliability of Provisional Responses in Session Initiation Protocol (SIP)", RFC 3262, DOI 10.17487/RFC3262, June 2002, <<https://www.rfc-editor.org/info/rfc3262>>.
- [RFC3428] Campbell, B., Ed., Rosenberg, J., Schulzrinne, H., Huitema, C., and D. Gurle, "Session Initiation Protocol (SIP) Extension for Instant Messaging", RFC 3428, DOI 10.17487/RFC3428, December 2002, <<https://www.rfc-editor.org/info/rfc3428>>.
- [RFC4119] Peterson, J., "A Presence-based GEOPRIV Location Object Format", RFC 4119, DOI 10.17487/RFC4119, December 2005, <<https://www.rfc-editor.org/info/rfc4119>>.
- [RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <<https://www.rfc-editor.org/info/rfc5234>>.
- [RFC7303] Thompson, H. and C. Lilley, "XML Media Types", RFC 7303, DOI 10.17487/RFC7303, July 2014, <<https://www.rfc-editor.org/info/rfc7303>>.
- [RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, DOI 10.17487/RFC3629, November 2003, <<https://www.rfc-editor.org/info/rfc3629>>.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <<https://www.rfc-editor.org/info/rfc3986>>.
- [RFC6442] Polk, J., Rosen, B., and J. Peterson, "Location Conveyance for the Session Initiation Protocol", RFC 6442, DOI 10.17487/RFC6442, December 2011, <<https://www.rfc-editor.org/info/rfc6442>>.
- [RFC6881] Rosen, B. and J. Polk, "Best Current Practice for Communications Services in Support of Emergency Calling", BCP 181, RFC 6881, DOI 10.17487/RFC6881, March 2013, <<https://www.rfc-editor.org/info/rfc6881>>.
- [RFC7852] Gellens, R., Rosen, B., Tschofenig, H., Marshall, R., and J. Winterbottom, "Additional Data Related to an Emergency Call", RFC 7852, DOI 10.17487/RFC7852, July 2016, <<https://www.rfc-editor.org/info/rfc7852>>.

- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8225] Wendt, C. and J. Peterson, "PASSporT: Personal Assertion Token", RFC 8225, DOI 10.17487/RFC8225, February 2018, <<https://www.rfc-editor.org/info/rfc8225>>.

## 12.2. Informative References

- [RFC7378] Tschofenig, H., Schulzrinne, H., and B. Aboba, Ed., "Trustworthy Location", RFC 7378, DOI 10.17487/RFC7378, December 2014, <<https://www.rfc-editor.org/info/rfc7378>>.
- [RFC8224] Peterson, J., Jennings, C., Rescorla, E., and C. Wendt, "Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 8224, DOI 10.17487/RFC8224, February 2018, <<https://www.rfc-editor.org/info/rfc8224>>.
- [RFC5031] Schulzrinne, H., "A Uniform Resource Name (URN) for Emergency and Other Well-Known Services", RFC 5031, DOI 10.17487/RFC5031, January 2008, <<https://www.rfc-editor.org/info/rfc5031>>.
- [RFC3325] Jennings, C., Peterson, J., and M. Watson, "Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks", RFC 3325, DOI 10.17487/RFC3325, November 2002, <<https://www.rfc-editor.org/info/rfc3325>>.
- [RFC5222] Hardie, T., Newton, A., Schulzrinne, H., and H. Tschofenig, "LoST: A Location-to-Service Translation Protocol", RFC 5222, DOI 10.17487/RFC5222, August 2008, <<https://www.rfc-editor.org/info/rfc5222>>.
- [RFC6443] Rosen, B., Schulzrinne, H., Polk, J., and A. Newton, "Framework for Emergency Calling Using Internet Multimedia", RFC 6443, DOI 10.17487/RFC6443, December 2011, <<https://www.rfc-editor.org/info/rfc6443>>.

## Authors' Addresses

Brian Rosen  
470 Conrad Dr  
Mars, PA 16046  
US

Phone:  
Email: [br@brianrosen.net](mailto:br@brianrosen.net)

Henning Schulzrinne  
Columbia University  
Department of Computer Science  
450 Computer Science Building  
New York, NY 10027  
US

Phone: +1 212 939 7004  
Email: [hgs+ecrit@cs.columbia.edu](mailto:hgs+ecrit@cs.columbia.edu)  
URI: <http://www.cs.columbia.edu>

Hannes Tschofenig  
ARM Limited

Austria

Email: [Hannes.Tschofenig@gmx.net](mailto:Hannes.Tschofenig@gmx.net)  
URI: <http://www.tschofenig.priv.at>

Randall Gellens  
Core Technology Consulting

Email: [rg+ietf@coretechnologyconsulting.com](mailto:rg+ietf@coretechnologyconsulting.com)  
URI: <http://www.coretechnologyconsulting.com>



ecrit  
Internet-Draft  
Intended status: BCP  
Expires: March 10, 2012

B. Rosen  
NeuStar  
J. Polk  
Cisco Systems  
September 7, 2011

Best Current Practice for Communications Services in support of  
Emergency Calling  
draft-ietf-ecrit-phonebcpr-20.txt

Abstract

The IETF and other standards organization have efforts targeted at standardizing various aspects of placing emergency calls on IP networks. This memo describes best current practice on how devices, networks and services using IETF protocols should use such standards to make emergency calls.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 10, 2012.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Terminology . . . . .	4
2. Introduction . . . . .	4
3. Overview of how emergency calls are placed . . . . .	4
4. Which devices and services should support emergency calls . . . . .	5
5. Identifying an emergency call . . . . .	5
6. Location and its role in an emergency call . . . . .	6
6.1. Types of location information . . . . .	7
6.2. Location Determination . . . . .	7
6.2.1. User-entered location information . . . . .	7
6.2.2. Access network "wire database" location information . . . . .	7
6.2.3. End-system measured location information . . . . .	8
6.2.4. Network-measured location information . . . . .	8
6.3. Who adds location, endpoint or proxy . . . . .	9
6.4. Location and references to location . . . . .	9
6.5. End system location configuration . . . . .	9
6.6. When location should be configured . . . . .	10
6.7. Conveying location . . . . .	11
6.8. Location updates . . . . .	12
6.9. Multiple locations . . . . .	12
6.10. Location validation . . . . .	13
6.11. Default location . . . . .	13
6.12. Other location considerations . . . . .	13
7. LIS and LoST Discovery . . . . .	14
8. Routing the call to the PSAP . . . . .	14
9. Signaling of emergency calls . . . . .	15
9.1. Use of TLS . . . . .	15
9.2. SIP signaling requirements for User Agents . . . . .	16
9.3. SIP signaling requirements for proxy servers . . . . .	17
10. Call backs . . . . .	18
11. Mid-call behavior . . . . .	18
12. Call termination . . . . .	18
13. Disabling of features . . . . .	18
14. Media . . . . .	19
15. Testing . . . . .	20
16. Security Considerations . . . . .	21
17. IANA Considerations . . . . .	21
17.1. test service urn . . . . .	21
17.2. 'test' Subregistry . . . . .	21
18. Acknowledgements . . . . .	22
19. References . . . . .	22
19.1. Normative References . . . . .	22
19.2. Informative References . . . . .	25

Authors' Addresses . . . . .	26
------------------------------	----

## 1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

This document uses terms from [RFC3261], [RFC5012] and [I-D.ietf-ecrit-framework].

## 2. Introduction

This document describes how access networks, Session Initiation Protocol [RFC3261] user agents, proxy servers and Public Safety Access Points (PSAPs) support emergency calling, as outlined in [I-D.ietf-ecrit-framework], which is designed to complement the present document in section headings, numbering and content. Understanding [I-D.ietf-ecrit-framework] is necessary to understand this document. This BCP succinctly describes the requirements of end devices and applications (requirements prefaced by "ED-"), access networks (including enterprise access networks) (requirements prefaced by "AN-"), service providers (requirements prefaced by "SP-") and PSAPs to achieve globally interoperable emergency calling on the Internet.

This document also defines requirements for "Intermediate" devices which exist between end devices or applications and the access network. For example, a home router is an "Intermediate" device. Reporting location on an emergency call (see Section 6) may depend on the ability of such intermediate devices to meet the requirements prefaced by "INT-".

The access network requirements apply to those networks which may be used to place emergency calls using IETF protocols. Local regulations may impact the need to support this document's access network requirements.

Other organizations, such as the North American Emergency Number Association (NENA), define the PSAP interface. NENA's documents reference this document.

## 3. Overview of how emergency calls are placed

An emergency call can be distinguished (Section 5) from any other call by a unique Service URN [RFC5031], which is placed in the call set-up signaling when a home or visited emergency dial string is

detected. Because emergency services are local to specific geographic regions, a caller must obtain his location (Section 6) prior to making emergency calls. To get this location, either a form of measuring (e.g., GPS) (Section 6.2.3) device location in the endpoint is deployed, or the endpoint is configured (Section 6.5) with its location from the access network's Location Information Server (LIS). The location is conveyed (Section 6.7) in the SIP signaling with the call. The call is routed (Section 8) based on location using the Location-to-Service Translation (LoST) protocol [RFC5222], which maps a location to a set of PSAP URIs. Each URI resolves to a PSAP or an Emergency Services Routing Proxy (ESRP), which serves a group of PSAPs. The call arrives at the PSAP with the location included in the SIP INVITE request.

#### 4. Which devices and services should support emergency calls

ED-1 A device or application that implements SIP calling SHOULD support emergency calling. Some jurisdictions have regulations governing which devices need to support emergency calling and developers are encouraged to ensure that devices they develop meet relevant regulatory requirements. Unfortunately, the natural variation in those regulations also makes it impossible to accurately describe the cases when developers do or do not have to support emergency calling.

SP-1 If a device or application expects to be able to place a call for help, the service provider that supports it MUST facilitate emergency calling. Some jurisdictions have regulations governing this.

ED-2 Devices that create media sessions and exchange real-time audio, video and/or text, have the capability to establish sessions to a wide variety of addresses, and communicate over private IP networks or the Internet, SHOULD support emergency calls. Some jurisdictions have regulations governing this.

#### 5. Identifying an emergency call

ED-3 Endpoints SHOULD recognize dial strings of emergency calls. If the service provider always knows the location of the device (the correct dial string depends on which country you are in), the service provider may recognize them, see SP-2.

SP-2 Proxy servers SHOULD recognize emergency dial strings if for some reason the endpoint does not recognize them.

ED-4/SP-3 Emergency calls MUST be marked with a Service URN in the Request-URI of the INVITE.

ED-5/SP-4 Geographically local dial strings MUST be recognized.

ED-6/SP-5 Devices MUST be able to be configured with the home country from which the home dial string(s) can be determined.

ED-7/SP-6 Emergency dial strings SHOULD be determined from LoST [RFC5222]. Dial Strings MAY be configured directly into the device.

AN-1 LoST servers MUST return dial strings for emergency services.

ED-8 Endpoints which do not recognize emergency dial strings SHOULD send dial strings as per [RFC4967].

SP-7 If a proxy server recognizes dial strings on behalf of its clients, it MUST recognize emergency dial strings represented by [RFC4967] and SHOULD recognize the emergency dial strings represented by a tel URI [RFC3966].

ED-9 Endpoints SHOULD be able to have home dial strings provisioned.

SP-8 Service providers MAY provision home dial strings in devices.

ED-10 Devices SHOULD NOT have one button emergency calling initiation.

ED-11/SP-9 All sub-services for the 'sos' service specified in [RFC5031]. MUST be recognized.

## 6. Location and its role in an emergency call

Handling location for emergency calling usually involves several steps to process and multiple entities are involved. In Internet emergency calling, where the endpoint is located is "determined" using a variety of measurement or wiretracing methods. Endpoints can be "configured" with their own location by the access network. In some circumstances, a proxy server can insert location into the signaling on behalf of the endpoint. The location is "mapped" to the URI to send the call to, and the location is "conveyed" to the PSAP (and other entities) in the signaling. Likewise, we employ Location Configuration Protocols (LCPs), the Location-to-Service Mapping Protocol, and Location Conveyance Protocols for these functions. The Location-to-Service Translation protocol [RFC5222] is the Location Mapping Protocol defined by the IETF.

### 6.1. Types of location information

There are several forms of location. All IETF location configuration and location conveyance protocols support both civic and geospatial (geo) forms. The civic forms include both postal and jurisdictional fields. A cell tower/sector can be represented as a point (geo or civic) or polygon. Endpoints, Intermediate Devices and Service Providers receiving other forms of location representation MUST map them into either a geo or civic for use in emergency calls.

ED-12/INT-1/SP-10 Endpoints, Intermediate Devices and Service Providers MUST be prepared to handle location represented in either civic or geo form.

ED-13/INT-2/SP-11/AN-2 Entities MUST NOT convert (civic to geo or geo to civic) from the form of location the determination mechanism (see Section Section 6.2) supplied prior to receipt by the PSAP.

### 6.2. Location Determination

ED-14/INT-3/AN-3 Any location determination mechanism MAY be used, provided the accuracy of the location meets local requirements.

#### 6.2.1. User-entered location information

ED-15/INT-4/AN-4 Devices, intermediate Devices and/or access networks SHOULD support a manual method to override the location the access network determines. When the override location is supplied in civic form, it MUST be possible for the resultant Presence Information Data Format - Location Object (PIDF-LO) received at the PSAP to contain any of the elements specified in [RFC4119] and [RFC5139].

#### 6.2.2. Access network "wire database" location information

AN-5 Access networks supporting copper, fiber or other hard wired IP packet service SHOULD support location configuration. If the network does not support location configuration, it MUST require every device or intermediate device that connects to the network to support end system measured location.

AN-6/INT-5 Access networks and intermediate devices providing wire database location information SHOULD provide interior location data (building, floor, room, cubicle) where possible. It is RECOMMENDED that interior location be provided when spaces exceed approximately 650 square meters. See [I-D.ietf-ecrit-framework] Section 6.2.2 for a discussion of how this value was determined.

AN-7/INT-6 Access networks and intermediate devices (including

enterprise networks) which support intermediate range wireless connections (typically 100m or less of range) and which do not support a more accurate location determination mechanism such as triangulation, MUST support location configuration where the location of the access point is reflected as the location of the clients of that access point.

AN-8/INT-7 Where the access network provides location configuration, intermediate devices MUST either be transparent to it, or provide an interconnected client for the supported configuration mechanism and a server for a configuration protocol supported by end devices downstream of the intermediate device such that the location provided by the access network is available to clients as if the intermediate device was not in the path.

#### 6.2.3. End-system measured location information

ED-16/INT-8 Devices MAY support end-system measured location. See [I-D.ietf-ecrit-framework] Section 6 for a discussion of accuracy of location.

ED-17/INT-9/AN-9 Devices that support endpoint measuring of location MUST have at least a coarse location capability (typically <1km accuracy) for routing of calls. The location mechanism MAY be a service provided by the access network.

#### 6.2.4. Network-measured location information

AN-10 Access networks MAY provide network-measured location determination. Wireless access networks that do not supply network measured location MUST require every device or intermediate device connected to the network to support end-system measured location. Uncertainty and confidence may be specified by local regulation. Where not specified, uncertainty of less than 100 meters with 95% confidence is RECOMMENDED for dispatch location.

AN-11 Access networks that provide network measured location MUST have at least a coarse location (typically <1km when not location hiding) capability at all times for routing of calls.

AN-12 Access networks with range of <10 meters (e.g. personal area networks such as Bluetooth MUST provide a location to mobile devices connected to them. The location provided SHOULD be that reported by the upstream access network unless a more accurate mechanism is available.



### 6.3. Who adds location, endpoint or proxy

ED-18/INT-10 Endpoints SHOULD attempt to configure their own location using the Location Configuration Protocols (LCPs) listed in ED-21.

SP-12 Proxies MAY provide location on behalf of devices if:

- o The proxy has a relationship with all access networks the device could connect to, and the relationship allows it to obtain location.
- o The proxy has an identifier, such as an IP address, that can be used by the access network to determine the location of the endpoint, even in the presence of NAT and VPN tunnels that may obscure the identifier between the access network and the service provider.

ED-19/INT-11/SP-13 Where proxies provide location on behalf of endpoints, the service provider MUST ensure that either the end device is provided with the local dial strings for its current location (where the end device recognizes dial strings), or the service provider proxy MUST detect the appropriate local dial strings at the time of the call.

### 6.4. Location and references to location

ED-20/INT-12 Devices SHOULD be able to accept and forward location by value or by reference. An end device that receives location by reference (and does not also get the corresponding value) MUST be able to perform a dereference operation to obtain a value.

### 6.5. End system location configuration

Obtaining location from the access network may be preferable even if the device can measure its own location, especially indoors where most measurement mechanisms are not accurate enough. This sections requirements do not apply to devices that can accurately measure their own location.

ED-21/INT-13 Devices MUST support both the Dynamic Host Configuration Protocol (DHCP) location options [RFC4776], [RFC6225] and HTTP Enabled Location Delivery (HELD) [RFC5985]. When devices deploy a specific access network interface for which location configuration mechanisms such as Link Layer Discovery Protocol - Media Endpoint Discovery (LLDP-MED) [LLDP-MED] or 802.11v are specified, the device SHOULD support the additional respective access network specific location configuration mechanism.

AN-13/INT-14 The access network MUST support either DHCP location options or HELD. The access network SHOULD support other location

configuration technologies that are specific to the type of access network.

AN-14/INT-15 Where a router is employed between a LAN and WAN in a small (less than approximately 650 square meters) area, the router MUST be transparent to the location provided by the WAN to the LAN. This may mean the router must obtain location as a client from the WAN, and supply an LCP server to the LAN with the location it obtains. Where the area is larger, the LAN MUST have a location configuration mechanism satisfying the requirements of this document.

ED-22/INT-16 Endpoints SHOULD try all LCPs supported by the device in any order or in parallel. The first one that succeeds in supplying location MUST be used.

AN-15/INT-17 Access networks that support more than one LCP MUST reply with the same location information (within the limits of the data format for the specific LCP) for all LCPs it supports.

ED-23/INT-18/SP-14 When HELD is the LCP, the request MUST specify a value of "emergencyRouting" for the "responseTime" parameter and use the resulting location for routing. If a value for dispatch location will be sent, another request with the "responseTime" parameter set to "emergencyDispatch" must be completed, with the result sent for dispatch purposes.

ED-24 Where the operating system supporting application programs which need location for emergency calls does not allow access to Layer 2 and Layer 3 functions necessary for a client application to use DHCP location options and/or other location technologies that are specific to the type of access network, the operating system MUST provide a published API conforming to ED-12 through ED-23 and ED-25 through ED-32. It is RECOMMENDED that all operating systems provide such an API.

#### 6.6. When location should be configured

If an endpoint is manually configured, the requirements in this section are not applicable.

ED-25/INT-19 Endpoints SHOULD obtain location immediately after obtaining local network configuration information.

ED-26/INT-20 If the device is configured to use DHCP for bootstrapping, and does not use it MUST include both options for location acquisition (civic and geodetic), the option for LIS discovery, and the option for LoST discovery as defined in [RFC4776], [RFC6225], [RFC5986] and [RFC5223].

ED-27/INT-21 If the device sends a DHCPINFORM message, it MUST include both options for location acquisition (civic and geodetic), the option for LIS discovery, and the option for LoST discovery as defined in [RFC4776], [RFC6225], [RFC5986] and [RFC5223].

ED-28/INT-22 To minimize the effects of VPNs that do not allow packets to be sent via the native hardware interface rather than via the VPN tunnel, location configuration SHOULD be attempted before such tunnels are established.

ED-29/INT-23 Software which uses LCPs SHOULD locate and use the actual hardware network interface rather than a VPN tunnel interface to direct LCP requests to the LIS in the actual access network.

AN-16 Network administrators MUST take care in assigning IP addresses such that VPN address assignments can be distinguished from local devices (by subnet choice, for example), and LISs SHOULD NOT attempt to provide location to addresses that arrive via VPN connections unless it can accurately determine the location for such addresses.

AN-17 Placement of NAT devices where an LCP uses IP address for an identifier SHOULD consider the effect of the NAT on the LCP. The address used to query the LIS MUST be able to correctly identify the record in the LIS representing the location of the querying device

ED-30/INT-24 For devices which are not expected to change location, refreshing location on the order of once per day is RECOMMENDED.

ED-31/INT-25 For devices which roam, refresh of location information SHOULD be more frequent, with the frequency related to the mobility of the device and the ability of the access network to support the refresh operation. If the device detects a link state change that might indicate having moved, for example when it changes access points, the device SHOULD refresh its location.

ED-32/INT-26/AN-18 It is RECOMMENDED that location determination not take longer than 250 ms to obtain routing location and systems SHOULD be designed such that the typical response is under 100 ms. However, as much as 3 seconds to obtain routing location MAY be tolerated if location accuracy can be substantially improved over what can be obtained in 250 ms.

## 6.7. Conveying location

ED-33/SP-15 Location sent between SIP entities MUST be conveyed using [I-D.ietf-sipcore-location-conveyance].

## 6.8. Location updates

ED-34/AN-19 Where the absolute location or the accuracy of location of the endpoint may change between the time the call is received at the PSAP and the time dispatch is completed, location update mechanisms **MUST** be implemented and used.

ED-35/AN-20 Mobile devices **MUST** be provided with a mechanism to get repeated location updates to track the motion of the device during the complete processing of the call.

ED-36/AN-21 The LIS **SHOULD** provide a location reference which permits a subscription with appropriate filtering.

ED-37/AN-22 For calls sent with location-by-reference, with a SIP or SIPS scheme, the server resolving the reference **MUST** support a SUBSCRIBE [RFC3265] to the presence event [RFC3856]. For other location-by-reference schemes that do not support subscription, the PSAP will have to repeatedly dereference the URI to determine if the device moved.

ED-38 If location was sent by value, and the endpoint gets updated location, it **MUST** send the updated location to the PSAP via a SIP re-INVITE or UPDATE request. Such updates **SHOULD** be limited to no more than one update every 10 seconds, a value selected to keep the load on a large PSAP manageable, and yet provide sufficient indication to the PSAP of motion.

## 6.9. Multiple locations

ED-39/SP-16 If the LIS has more than one location for an endpoint it **MUST** conform to the rules in Section 3 of [RFC5491]

ED-40 If an endpoint has more than one location available to it, it **MUST** choose one location to route the call towards the PSAP. If multiple locations are in a single Presence Information Data Format (PIDF), the procedures in [RFC5491] **MUST** be followed. If the endpoint has multiple PIDFs, and has no reasonable basis to choose from among them, a random choice is acceptable.

SP-17 If a proxy inserts location on behalf of an endpoint, and it has multiple locations available for the endpoint it **MUST** choose one location to use to route the call towards the PSAP. If multiple locations are in a single PIDF, the procedures in [RFC5491] **MUST** be followed. If the proxy has multiple PIDFs, and has no reasonable basis to choose from among them, a random choice is acceptable.

SP-18 If a proxy is attempting to insert location but the endpoint

conveyed a location to it, the proxy MUST use the endpoint's location for routing in the initial INVITE and MUST convey that location towards the PSAP. It MAY also include what it believes the location to be in a separate Geolocation header.

SP-19 All location objects received by a proxy MUST be delivered to the PSAP.

ED-41/SP-20 Location objects MUST be created with information about the method by which the location was determined, such as GPS, manually entered, or based on access network topology included in a PIDF-LO "method" element. In addition, the source of the location information MUST be included in a PIDF-LO "provided-by" element.

ED-42/SP-21 A location with a method of "derived" MUST NOT be used unless no other location is available.

#### 6.10. Location validation

AN-23 A LIS SHOULD perform location validation of civic locations via LoST before entering a location in its database.

ED-44 Endpoints SHOULD validate civic locations when they receive them from their LCP. Validation SHOULD be performed in conjunction with the LoST route query to minimize load on the LoST server.

#### 6.11. Default location

AN-24 When the access network cannot determine the actual location of the caller, it MUST supply a default location. The default SHOULD be chosen to be as close to the probable location of the device as the network can determine. See [I-D.ietf-ecrit-framework]

SP-22 Proxies handling emergency calls MUST insert a default location in the INVITE if the incoming INVITE does not contain a location and the proxy does not have a method for obtaining a better location.

AN-25/SP-23 Default locations MUST be marked with method=Default and the proxy MUST be identified in provided-by element of the PIDF-LO.

#### 6.12. Other location considerations

ED-45 If the LCP does not return location in the form of a PIDF-LO [RFC4119], the endpoint MUST map the location information it receives from the configuration protocol to a PIDF-LO.

ED-46/AN-26 To prevent against spoofing of the DHCP server, entities implementing DHCP for location configuration SHOULD use [RFC3118],

although the difficulty in providing appropriate credentials is significant.

ED-47 If S/MIME [RFC5751] is used, the INVITE message MUST provide enough information unencrypted for intermediate proxies to route the call based on the location information included. This would include the Geolocation header, and any bodies containing location information. Use of S/MIME with emergency calls is NOT RECOMMENDED for this reason.

ED-48/SP-24 TLS [RFC5746] MUST be used to protect location (but see Section 9.1). All implementations MUST support TLS.

## 7. LIS and LoST Discovery

ED-49 Endpoints MUST support one or more mechanisms that allow them to determine their public IP address, for example, STUN [RFC5389].

ED-50 Endpoints MUST support LIS discovery as described in [RFC5986], and the LoST discovery as described in [RFC5223].

ED-51 The device MUST have a configurable default LoST server parameter.

ED-52 DHCP LoST discovery MUST be used, if available, in preference to configured LoST servers. That is, the endpoint MUST send queries to this LoST server first, using other LoST servers only if these queries fail.

AN-27 Access networks which support DHCP MUST implement the LIS and LoST discovery options in their DHCP servers and return suitable server addresses as appropriate.

## 8. Routing the call to the PSAP

ED-53 Endpoints who obtain their own location SHOULD perform LoST mapping to the PSAP URI.

ED-54 Mapping SHOULD be performed at boot time and whenever location changes beyond the service boundary obtained from a prior LoST mapping operation or the time-to-live value of that response has expired. The value MUST be cached for possible later use.

ED-55 The endpoint MUST attempt to update its location at the time of an emergency call. If it cannot obtain a new location quickly (see Section 6), it MUST use the cached value.

ED-56 The endpoint SHOULD attempt to update the LoST mapping at the time of an emergency call. If it cannot obtain a new mapping quickly, it MUST use the cached value. If the device cannot update the LoST mapping and does not have a cached value, it MUST signal an emergency call without a Route header containing a PSAP URI.

SP-25 Networks MUST be designed so that at least one proxy in the outbound path will recognize emergency calls with a Request URI of the service URN in the "sos" tree. An endpoint places a service URN in the Request URI to indicate that the endpoint understood the call was an emergency call. A proxy that processes such a call looks for the presence of a SIP Route header field with a URI of a PSAP. Absence of such a Route header indicates the endpoint was unable to invoke LoST and the proxy MUST perform the LoST mapping and insert a Route header field with the URI obtained.

SP-26 To deal with old user agents that predate this specification and with endpoints that do not have access to their own location data, a proxy that recognizes a call as an emergency call that is not marked as such (see Section 5) MUST also perform this mapping, with the best location it has available for the endpoint. The resulting PSAP URI would be placed in a Route header with the service URN in the Request URI.

SP-27 Proxy servers performing mapping SHOULD use location obtained from the access network for the mapping. If no location is available, a default location (see Section 6.11) MUST be supplied.

SP-28 A proxy server which attempts mapping and fails to get a mapping MUST provide a default mapping. A suitable default mapping would be the mapping obtained previously for the default location appropriate for the caller.

ED-57/SP-29 [RFC3261] and [RFC3263] procedures MUST be used to route an emergency call towards the PSAP's URI.

## 9. Signaling of emergency calls

### 9.1. Use of TLS

ED-58/SP-30 TLS is the primary mechanism used to secure the signaling for emergency calls. IPsec [RFC4301] MAY be used instead of TLS for any hop. Either TLS or IPSEC MUST be used when attempting to signal an emergency call.

ED-59/SP-31 If TLS session establishment is not available, or fails, the call MUST be retried without TLS.

ED-60/SP-32 [RFC5626] is RECOMMENDED to maintain persistent TLS connections between entities when one of the entity is an endpoint. Persistent TLS connection between proxies is RECOMMENDED using any suitable mechanism.

ED-61/AN-28 TLS SHOULD be used when attempting to retrieve location (configuration or dereferencing) with HELD. The use of [RFC5077] is RECOMMENDED to minimize the time to establish TLS sessions without keeping server-side state. IPsec MAY be used instead of TLS.

ED-62/AN-29 When TLS session establishment fails, the location retrieval MUST be retried without TLS.

## 9.2. SIP signaling requirements for User Agents

ED-63 The initial SIP signaling method is an INVITE request:

1. The Request URI SHOULD be the service URN in the "sos" tree. If the device does not interpret local dial strings, the Request-URI MUST be a dial string URI [RFC4967] with the dialed digits.
2. The To header field SHOULD be a service URN in the "sos" tree. If the device does not interpret local dial strings, the To: MUST be a dial string URI with the dialed digits.
3. The From header field SHOULD contain the AoR of the caller.
4. A Route header field SHOULD be present with a PSAP URI obtained from LoST (see Section 8). If the device does not interpret dial plans, or was unable to obtain a route from a LoST server, no such Route header field will be present.
5. A Contact header field MUST be globally routable, for example a GRUU [RFC5627], and be valid for several minutes following the termination of the call, provided that the UAC remains registered with the same registrar, to permit an immediate call-back to the specific device which placed the emergency call. It is acceptable if the UAC inserts a locally routable URI and a subsequent B2BUA maps that to a globally routable URI.
6. Other header fields MAY be included as per normal SIP behavior.
7. If a geolocation URI is included in the INVITE, a Supported header field MUST be included with a 'geolocation-sip' or 'geolocation-http' option tag, as appropriate. [I-D.ietf-sipcore-location-conveyance].
8. If a device understands the SIP location conveyance [I-D.ietf-sipcore-location-conveyance] extension and has its location available, it MUST include location either by-value, by-reference or both.
9. A SDP offer SHOULD be included in the INVITE. If voice is supported the offer SHOULD include the G.711 codec, see Section 14. As PSAPs may support a wide range of media types and codecs, sending an offerless INVITE may result in a lengthy return offer, but is permitted. Cautions in [RFC3261] on



offerless INVITEs should be considered before such use.

10. If the device includes location-by-value, the UA MUST support multipart message bodies, since SDP will likely be also in the INVITE.

### 9.3. SIP signaling requirements for proxy servers

SP-33 SIP Proxy servers processing emergency calls:

1. If the proxy interprets dial plans on behalf of user agents, the proxy MUST look for the local emergency dial string at the location of the end device and MAY look for the home dial string. If it finds it, the proxy MUST:
  - \* Insert a Geolocation header field. Location-by-reference MUST be used because proxies must not insert bodies.
  - \* Insert the Geolocation-Routing header with appropriate parameters .
  - \* Map the location to a PSAP URI using LoST.
  - \* Add a Route header with the PSAP URI.
  - \* Replace the Request-URI (which was the dial string) with the service URN appropriate for the emergency dial string.
  - \* Route the call using normal SIP routing mechanisms.
2. If the proxy recognizes the service URN in the Request URI, and does not find a Route header, it MUST query a LoST server immediately. If a location was provided (which should be the case), the proxy uses that location to query LoST. The proxy may have to dereference a location by reference to get a value. If a location is not present, and the proxy can query a LIS which has the location of the UA it MUST do so. If no location is present, and the proxy does not have access to a LIS which could provide location, the proxy MUST supply a default location (See Section 6.11). The location (in the signaling, obtained from a LIS, or default) MUST be used in a query to LoST with the service URN received with the call. The resulting URI MUST be placed in a Route header added to the call.
3. The proxy MAY add a Geolocation header field. Such an additional location SHOULD NOT be used for routing; the location provided by the UA should be used.
4. Either a P-Asserted-Identity [RFC3325] or an Identity header field [RFC4474], or both, SHOULD be included to identify the sender. For services which must support emergency calls from unauthenticated devices, valid identity may not be available. Proxies encountering a P-Asserted-Identity will need to pass the header to the PSAP, which is in a different domain. [RFC3325] requires a "spec(T)" to determine what happens if the "id" privacy service, or a Privacy header is present and requests privacy. In the absence of another spec(T), such proxies should pass the header unmodified if and only if the connection between the proxy and the PSAP is, as far as the proxy can determine,

protected by TLS with mutual authentication using keys reliably known by the parties, encrypted with no less strength than AES and the local regulations governing the PSAP do not otherwise specify.

5. Proxies SHOULD NOT return a 424 error. It should process the INVITE as best as it can.
6. Proxies SHOULD NOT obey a Geolocation-Routing value of "no" or a missing value if the proxy must query LoST to obtain a route. Emergency calls are always routed by location.

## 10. Call backs

ED-64/SP-34 Devices device SHOULD have a globally routable URI in a Contact: header field which remains valid for several minutes past the time the original call containing the URI completes unless the device registration expires and is not renewed.

SP-35 Call backs to the Contact: header URI received within 30 minutes of an emergency call must reach the device regardless of call features or services that would normally cause the call to be routed to some other entity.

SP-36 Devices MUST have a persistent AOR URI either in a P-Asserted-Identity header field or From protected by an Identity header field suitable for returning a call some time after the original call. Such a call back would not necessarily reach the device that originally placed the call.

## 11. Mid-call behavior

ED-65/SP-37 During the course of an emergency call, devices and proxies MUST initiate a call transfer upon receipt of REFER request within the dialog with method=INVITE and the Referred-by header field [RFC3515] in that request.

## 12. Call termination

ED-66 Normal [RFC3261] procedures for termination MUST be used for termination of the call.

## 13. Disabling of features

ED-67/SP-38 User Agents and proxies MUST disable features that will interrupt an ongoing emergency call, such as:

- o Call Waiting
- o Call Transfer
- o Three Way Call
- o Hold
- o Outbound Call Blocking

when an emergency call is established, but see ED-66 with respect to Call Waiting. Also see ED-74 in Section 14.

ED-68/SP-39 The emergency dial strings SHOULD NOT be permitted in Call Forward numbers or speed dial lists.

ED-69/SP-40 The User Agent and Proxies MUST disable call features which would interfere with the ability of call backs from the PSAP to be completed such as:

- o Do Not Disturb
- o Call Forward (all kinds)

These features SHOULD be disabled for approximately 30 minutes following termination of an emergency call.

ED-70 Call backs SHOULD be determined by retaining the domain of the PSAP which answers an outgoing emergency call and instantiating a timer which starts when the call is terminated. If a call is received from the same domain and within the timer period, sent to the Contact: or AoR used in the emergency call, it should be assumed to be a call back. The suggested timer period is 5 minutes. [RFC4916] may be used by the PSAP to inform the endpoint of the domain of the PSAP. Recognizing a call back from the domain of the PSAP will not always work, and further standardization will be required to give the endpoint the ability to recognize a call back.

#### 14. Media

ED-71 Endpoints MUST send and receive media streams on RTP [RFC3550].

ED-72 Normal SIP offer/answer [RFC3264] negotiations MUST be used to agree on the media streams to be used.

ED-73/SP-41 G.711 A law (and mu Law if they are intended be used in North America) encoded voice as described in [RFC3551] MUST be supported. If the endpoint cannot support G.711, a transcoder MUST be used so that the offer received at the PSAP contains G.711. It is desirable to include wideband codecs such as G.722 and AMR-WB in the offer. PSAPs SHOULD support narrowband codecs common on endpoints in their area to avoid transcoding.

ED-74 Silence suppression (Voice Activity Detection methods) MUST NOT be used on emergency calls. PSAP call takers sometimes get

information on what is happening in the background to determine how to process the call.

ED-75 Endpoints supporting Instant Messaging (IM) MUST support either [RFC3428] and [RFC4975].

ED-76 Endpoints supporting real-time text MUST use [RFC4103]. The expectations for emergency service support for the real-time text medium are described in [RFC5194], Section 7.1.

ED-77 Endpoints supporting video MUST support H.264 per [RFC6184].

## 15. Testing

ED-78 INVITE requests to a service URN starting with "test." indicates a request for an automated test. For example, "urn:service:test.sos.fire". As in standard SIP, a 200 (OK) response indicates that the address was recognized and a 404 (Not found) that it was not. A 486 (Busy Here) MUST be returned if the test service is busy, and a 404 (Not found) MUST be returned if the PSAP does not support the test mechanism.

ED-79 In its response to the test, the PSAP MAY include a text body (text/plain) indicating the identity of the PSAP, the requested service, and the location reported with the call. For the latter, the PSAP SHOULD return location-by-value even if the original location delivered with the test was by-reference. If the location-by-reference was supplied, and the dereference requires credentials, the PSAP SHOULD use credentials supplied by the LIS for test purposes. This alerts the LIS that the dereference is not for an actual emergency call and location hiding techniques, if they are being used, may be employed for this dereference. Use of SIPS for the request would assure the response containing the location is kept private

ED-80 A PSAP accepting a test call SHOULD accept a media loopback test [I-D.ietf-mmusic-media-loopback] and SHOULD support the "rtp-pkt-loopback" and "rtp-start-loopback" options. The user agent would specify a loopback attribute of "loopback-source", the PSAP being the mirror. User Agents should expect the PSAP to loop back no more than 3 packets of each media type accepted (which limits the duration of the test), after which the PSAP would normally send BYE.

ED-81 User agents SHOULD perform a full call test, including media loopback, after a disconnect and subsequent change in IP address not due to a reboot. After an initial test, a full test SHOULD be repeated approximately every 30 days with a random interval.

ED-82 User agents MUST NOT place a test call immediately after booting. If the IP address changes after booting, the endpoint should wait a random amount of time (in perhaps a 30 minute period, sufficient for any avalanche restart to complete) and then test.

ED-83 PSAPs MAY refuse repeated requests for test from the same device in a short period of time. Any refusal is signaled with a 486 or 488 response.

## 16. Security Considerations

Security considerations for emergency calling have been documented in [RFC5069], and [RFC6280]. This document suggests that security (TLS or IPsec) be used hop by hop on a SIP call to protect location information, identity, etc. It also suggests that if the attempt to create a security association fails, the call be retried without the security. It's more important to get an emergency call through than to protect the data; indeed, in many jurisdictions privacy is explicitly waived when making emergency calls. Placing a call without security may reveal user information, including location. The alternative - failing the call if security cannot be established, is considered unacceptable.

## 17. IANA Considerations

This document registers service URNs in the Service URN Labels registry per [RFC5031] for testing.

### 17.1. test service urn

A new entry in the URN Service Label registry is added. The new service is "test", the reference is this document, and the description is "self test".

### 17.2. 'test' Subregistry

A new Subregistry is created, the "'test' Sub-Service. The registration process is Expert Review per [RFC5226]. The expert review should consider that the entries in this registry nominally track the entries in the sos sub registry, although it is not required that every entry in sos have an entry in test, and it is possible that entries in the test sub-registry not necessarily be in the sos sub registry. For example, testing of non-emergency URNs may be allowed. The Reference is this document. The initial content of the subregistry is:

Service	Reference	Description
test.sos	[this document]	test for sos
test.sos.ambulance	[this document]	test for sos.ambulance
test.sos.animal-control	[this document]	test for sos.animal-control
test.sos.fire	[this document]	test for sos.fire
test.sos.gas	[this document]	test for sos.gas
test.sos.marine	[this document]	test for sos.marine
test.sos.mountain	[this document]	test for sos.mountain
test.sos.physician	[this document]	test for sos.physician
test.sos.poison	[this document]	test for sos.poison
test.sos.police	[this document]	test for sos.police

## 18. Acknowledgements

Work group members participating in the creation and review of this document include Hannes Tschofenig, Ted Hardie, Marc Linsner, Roger Marshall, Stu Goldman, Shida Schubert, James Winterbottom, Barbara Stark, Richard Barnes and Peter Blatherwick.

## 19. References

### 19.1. Normative References

- [I-D.ietf-mmusic-media-loopback]  
Sivachelvan, C., Venna, N., Jones, P., Stratton, N., Roychowdhury, A., and K. Hedayat, "An Extension to the Session Description Protocol (SDP) for Media Loopback", draft-ietf-mmusic-media-loopback-15 (work in progress), March 2011.
- [I-D.ietf-sipcore-location-conveyance]  
Polk, J., Rosen, B., and J. Peterson, "Location Conveyance for the Session Initiation Protocol", draft-ietf-sipcore-location-conveyance-09 (work in progress), September 2011.
- [LLDP-MED]  
TIA, "ANSI/TIA-1057 Link Layer Discovery Protocol - Media Endpoint Discovery".
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3118] Droms, R. and W. Arbaugh, "Authentication for DHCP Messages", RFC 3118, June 2001.

- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.
- [RFC3263] Rosenberg, J. and H. Schulzrinne, "Session Initiation Protocol (SIP): Locating SIP Servers", RFC 3263, June 2002.
- [RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.
- [RFC3265] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002.
- [RFC3428] Campbell, B., Rosenberg, J., Schulzrinne, H., Huitema, C., and D. Gurle, "Session Initiation Protocol (SIP) Extension for Instant Messaging", RFC 3428, December 2002.
- [RFC3515] Sparks, R., "The Session Initiation Protocol (SIP) Refer Method", RFC 3515, April 2003.
- [RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.
- [RFC3551] Schulzrinne, H. and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control", STD 65, RFC 3551, July 2003.
- [RFC3856] Rosenberg, J., "A Presence Event Package for the Session Initiation Protocol (SIP)", RFC 3856, August 2004.
- [RFC3966] Schulzrinne, H., "The tel URI for Telephone Numbers", RFC 3966, December 2004.
- [RFC4103] Hellstrom, G. and P. Jones, "RTP Payload for Text Conversation", RFC 4103, June 2005.
- [RFC4119] Peterson, J., "A Presence-based GEOPRIV Location Object Format", RFC 4119, December 2005.
- [RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.
- [RFC4474] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session

Initiation Protocol (SIP)", RFC 4474, August 2006.

- [RFC4776] Schulzrinne, H., "Dynamic Host Configuration Protocol (DHCPv4 and DHCPv6) Option for Civic Addresses Configuration Information", RFC 4776, November 2006.
- [RFC4916] Elwell, J., "Connected Identity in the Session Initiation Protocol (SIP)", RFC 4916, June 2007.
- [RFC4967] Rosen, B., "Dial String Parameter for the Session Initiation Protocol Uniform Resource Identifier", RFC 4967, July 2007.
- [RFC4975] Campbell, B., Mahy, R., and C. Jennings, "The Message Session Relay Protocol (MSRP)", RFC 4975, September 2007.
- [RFC5031] Schulzrinne, H., "A Uniform Resource Name (URN) for Emergency and Other Well-Known Services", RFC 5031, January 2008.
- [RFC5139] Thomson, M. and J. Winterbottom, "Revised Civic Location Format for Presence Information Data Format Location Object (PIDF-LO)", RFC 5139, February 2008.
- [RFC5222] Hardie, T., Newton, A., Schulzrinne, H., and H. Tschofenig, "LoST: A Location-to-Service Translation Protocol", RFC 5222, August 2008.
- [RFC5223] Schulzrinne, H., Polk, J., and H. Tschofenig, "Discovering Location-to-Service Translation (LoST) Servers Using the Dynamic Host Configuration Protocol (DHCP)", RFC 5223, August 2008.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.
- [RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.
- [RFC5491] Winterbottom, J., Thomson, M., and H. Tschofenig, "GEOPRIV Presence Information Data Format Location Object (PIDF-LO) Usage Clarification, Considerations, and Recommendations", RFC 5491, March 2009.
- [RFC5626] Jennings, C., Mahy, R., and F. Audet, "Managing Client-Initiated Connections in the Session Initiation Protocol



(SIP)", RFC 5626, October 2009.

[RFC5627] Rosenberg, J., "Obtaining and Using Globally Routable User Agent URIs (GRUUs) in the Session Initiation Protocol (SIP)", RFC 5627, October 2009.

[RFC5746] Rescorla, E., Ray, M., Dispensa, S., and N. Oskov, "Transport Layer Security (TLS) Renegotiation Indication Extension", RFC 5746, February 2010.

[RFC5751] Ramsdell, B. and S. Turner, "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification", RFC 5751, January 2010.

[RFC5985] Barnes, M., "HTTP-Enabled Location Delivery (HELD)", RFC 5985, September 2010.

[RFC5986] Thomson, M. and J. Winterbottom, "Discovering the Local Location Information Server (LIS)", RFC 5986, September 2010.

[RFC6184] Wang, Y., Even, R., Kristensen, T., and R. Jesup, "RTP Payload Format for H.264 Video", RFC 6184, May 2011.

[RFC6225] Polk, J., Linsner, M., Thomson, M., and B. Aboba, "Dynamic Host Configuration Protocol Options for Coordinate-Based Location Configuration Information", RFC 6225, July 2011.

## 19.2. Informative References

[I-D.ietf-ecrit-framework]

Rosen, B., Schulzrinne, H., Polk, J., and A. Newton, "Framework for Emergency Calling using Internet Multimedia", draft-ietf-ecrit-framework-12 (work in progress), October 2010.

[RFC3325] Jennings, C., Peterson, J., and M. Watson, "Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks", RFC 3325, November 2002.

[RFC5012] Schulzrinne, H. and R. Marshall, "Requirements for Emergency Context Resolution with Internet Technologies", RFC 5012, January 2008.

[RFC5069] Taylor, T., Tschofenig, H., Schulzrinne, H., and M. Shanmugam, "Security Threats and Requirements for Emergency Call Marking and Mapping", RFC 5069,

January 2008.

- [RFC5077] Salowey, J., Zhou, H., Eronen, P., and H. Tschofenig, "Transport Layer Security (TLS) Session Resumption without Server-Side State", RFC 5077, January 2008.
- [RFC5194] van Wijk, A. and G. Gybels, "Framework for Real-Time Text over IP Using the Session Initiation Protocol (SIP)", RFC 5194, June 2008.
- [RFC6280] Barnes, R., Lepinski, M., Cooper, A., Morris, J., Tschofenig, H., and H. Schulzrinne, "An Architecture for Location and Location Privacy in Internet Applications", BCP 160, RFC 6280, July 2011.

#### Authors' Addresses

Brian Rosen  
NeuStar  
470 Conrad Dr.  
Mars, PA 16046  
USA

Phone: +1 724 382 1051  
Email: br@brianrosen.net

James Polk  
Cisco Systems  
3913 Treemont Circle  
Colleyville, TX 76034  
USA

Phone: +1-817-271-3552  
Email: jmpolk@cisco.com



ECRIT  
Internet-Draft  
Intended status: Standards Track  
Expires: April 17, 2014

H. Schulzrinne  
Columbia University  
H. Tschofenig  
Nokia Solutions and Networks  
C. Holmberg  
Ericsson  
M. Patel  
InterDigital Communications  
October 14, 2013

Public Safety Answering Point (PSAP) Callback  
draft-ietf-ecrit-psap-callback-13.txt

Abstract

After an emergency call is completed (either prematurely terminated by the emergency caller or normally by the call taker) it is possible that the call taker feels the need for further communication. For example, the call may have been dropped by accident without the call taker having sufficient information about the current situation of a wounded person. A call taker may trigger a callback towards the emergency caller using the contact information provided with the initial emergency call. This callback could, under certain circumstances, be treated like any other call and as a consequence it may get blocked by authorization policies or may get forwarded to an answering machine.

The IETF emergency services architecture specification already offers a solution approach for allowing PSAP callbacks to bypass authorization policies to reach the caller without unnecessary delays. Unfortunately, the specified mechanism only supports limited scenarios. This document discusses shortcomings of the current mechanisms and illustrates additional scenarios where better-than-normal call treatment behavior would be desirable. A solution based on a new header field value, called "psap-callback", for the SIP Priority header field is specified to accomplish the PSAP callback marking.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 17, 2014.

#### Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

#### Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	4
3. Callback Scenarios . . . . .	4
3.1. Routing Asymmetry . . . . .	5
3.2. Multi-Stage Routing . . . . .	5
3.3. Call Forwarding . . . . .	6
3.4. Network-based Service URN Resolution . . . . .	8
3.5. PSTN Interworking . . . . .	9
4. SIP PSAP Callback Indicator . . . . .	10
4.1. General . . . . .	10
4.2. Usage . . . . .	10
4.3. Syntax . . . . .	10
4.3.1. General . . . . .	10
4.3.2. ABNF . . . . .	10
5. Security Considerations . . . . .	10
5.1. Security Threat . . . . .	10
5.2. Security Requirements . . . . .	11
5.3. Security Solution . . . . .	11
6. IANA Considerations . . . . .	13
7. Acknowledgements . . . . .	13
8. References . . . . .	14
8.1. Normative References . . . . .	14
8.2. Informative References . . . . .	14

## 1. Introduction

Summoning police, the fire department or an ambulance in emergencies is one of the fundamental and most-valued functions of the telephone. As telephone functionality moves from circuit-switched telephony to Internet telephony, its users rightfully expect that this core functionality will continue to work at least as well as it has for the legacy technology. New devices and services are being made available that could be used to make a request for help, which are not traditional telephones, and users are increasingly expecting them to be used to place emergency calls.

An overview of the protocol interactions for emergency calling using the IETF emergency services architecture are described in [RFC6443] and [RFC6881] specifies the technical details. As part of the emergency call setup procedure two important identifiers are conveyed to the PSAP call taker's user agent, namely the Address-Of-Record (AOR), and, if available, the Globally Routable User Agent (UA) URIs (GRUU). RFC 3261 [RFC3261] defines the AOR as:

"An address-of-record (AOR) is a SIP or SIPS URI that points to a domain with a location service that can map the URI to another URI where the user might be available. Typically, the location service is populated through registrations. An AOR is frequently thought of as the "public address" of the user."

In SIP systems a single user can have a number of user agents (handsets, softphones, voicemail accounts, etc.) which are all referenced by the same AOR. There are a number of cases in which it is desirable to have an identifier which addresses a single user agent rather than the group of user agents indicated by an AOR. The GRUU is such a unique user-agent identifier, which is still globally routable. RFC 5627 [RFC5627] specifies how to obtain and use GRUUs. [RFC6881] also makes use of the GRUU for emergency calls.

Regulatory requirements demand that the emergency call setup procedure itself provides enough information to allow the call taker to initiate a callback to the emergency caller. This is desirable in those cases where the call got dropped prematurely or when further communication need arises. The AOR and the GRUU serve this purpose.

The communication attempt by the PSAP call taker back to the emergency caller is called 'PSAP callback'.

A PSAP callback may, however, be blocked by user configured authorization policies or may be forwarded to an answering machine since SIP entities (SIP proxies as well as the SIP user equipment itself) cannot differentiate the PSAP callback from any other SIP

call. "Call barring", "do not disturb", or "call diversion"(aka call forwarding) are features that prevent delivery of a call. It is important to note that these features may be implemented by SIP intermediaries as well as by the user agent.

Among the emergency services community there is the desire to offer PSAP callbacks a treatment such that chances are increased that it reaches the emergency caller. At the same time a design must deal with the negative side-effects of allowing certain calls to bypass call forwarding or other authorization policies. Ideally, the PSAP callback has to relate to an earlier emergency call that was made "not too long ago". An exact time interval is difficult to define in a global IETF standard due to the variety of national regulatory requirements but [RFC6881] suggests 30 minutes.

To nevertheless meet the needs from the emergency services community a basic mechanism for preferential treatment of PSAP callbacks was defined in Section 13 of [RFC6443]. The specification says:

"A UA may be able to determine a PSAP callback by examining the domain of incoming calls after placing an emergency call and comparing that to the domain of the answering PSAP from the emergency call. Any call from the same domain and directed to the supplied Contact header or AOR after an emergency call should be accepted as a callback from the PSAP if it occurs within a reasonable time after an emergency call was placed."

This approach mimics a stateful packet filtering firewall and is indeed helpful in a number of cases. It is also relatively simple to implement even though it requires call state to be maintained by the user agent as well as by SIP intermediaries. Unfortunately, the solution does not work in all deployment scenarios. In Section 3 we describe cases where the currently standardized approach is insufficient.

## 2. Terminology

Emergency services related terminology is borrowed from [RFC5012]. This includes terminology like emergency caller, user equipment, call taker, Emergency Service Routing Proxy (ESRP), and Public Safety Answering Point (PSAP).

## 3. Callback Scenarios

This section illustrates a number of scenarios where the currently specified solution, as specified in [RFC6881], for preferential treatment of callbacks fails. As explained in Section 1 a SIP entity examines an incoming PSAP callback by comparing the domain of the

PSAP with the destination domain of the outbound emergency call placed earlier.

### 3.1. Routing Asymmetry

In some deployment environments it is common to have incoming and outgoing SIP messaging routed through different SIP entities. Figure 1 shows this graphically whereby a VoIP provider uses different SIP proxies for inbound and for outbound call handling. Unless the two devices are synchronized, the callback hitting the inbound proxy would get treated like any other call since the emergency call established state information at the outbound proxy only.

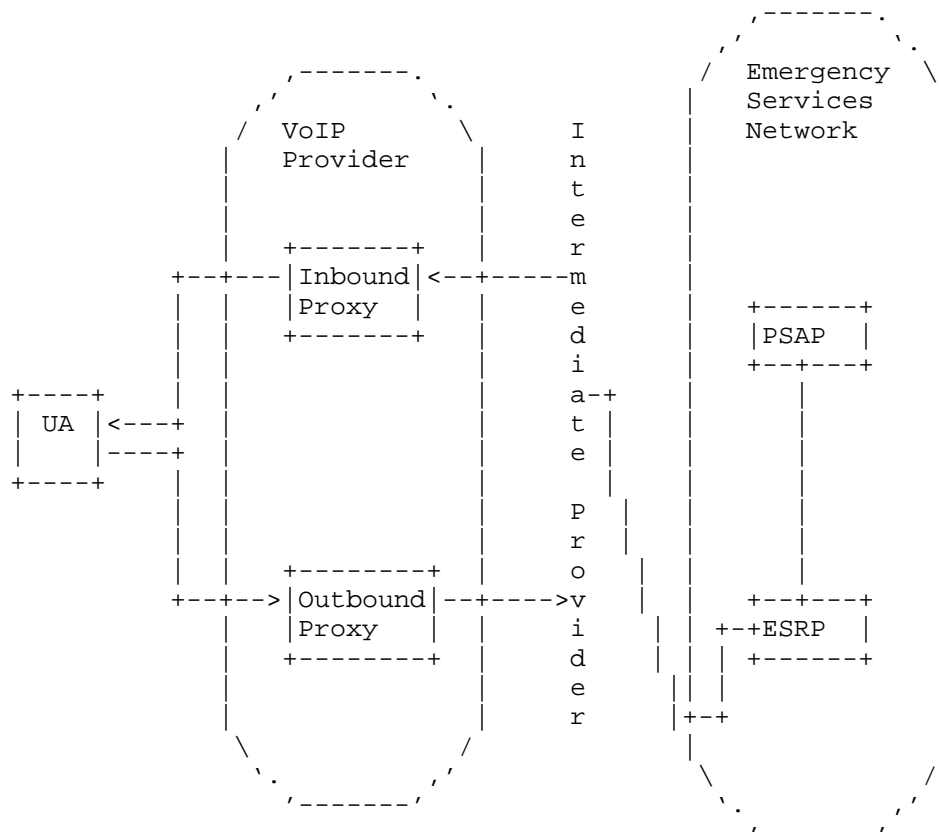


Figure 1: Example for Routing Asymmetry.

### 3.2. Multi-Stage Routing



Consider the following emergency call routing scenario shown in Figure 2 where routing towards the PSAP occurs in several stages. In this scenario we consider a SIP UA that uses the Location-to-Service Translation Protocol (LoST) [RFC5222] to learn the next hop destination, namely `esrp@example.net`, to get the call closer to the PSAP. This call is then sent to the proxy of the user's VoIP provider (`example.org`). The user's VoIP provider receives the emergency call and creates state based on the destination domain, namely `example.net`. It then routes it to the indicated ESRP. When the ESRP receives it it needs to decide what the next hop is to get to the final PSAP. In our example the next hop is the PSAP with the URI `psap@example.com`.

When a callback is sent from `psap@example.com` towards the emergency caller the call will get normal treatment by the proxy of the VoIP provider since the domain of the PSAP does not match the stored state information.

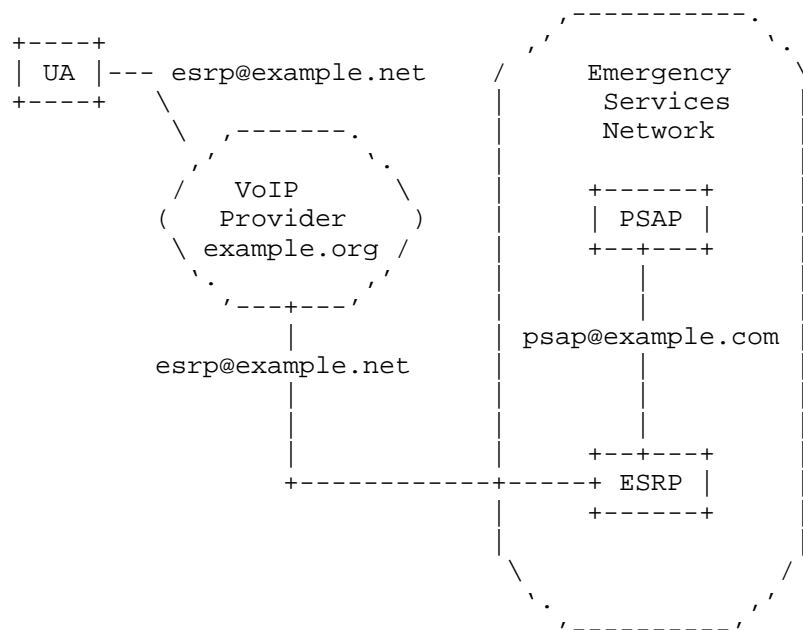


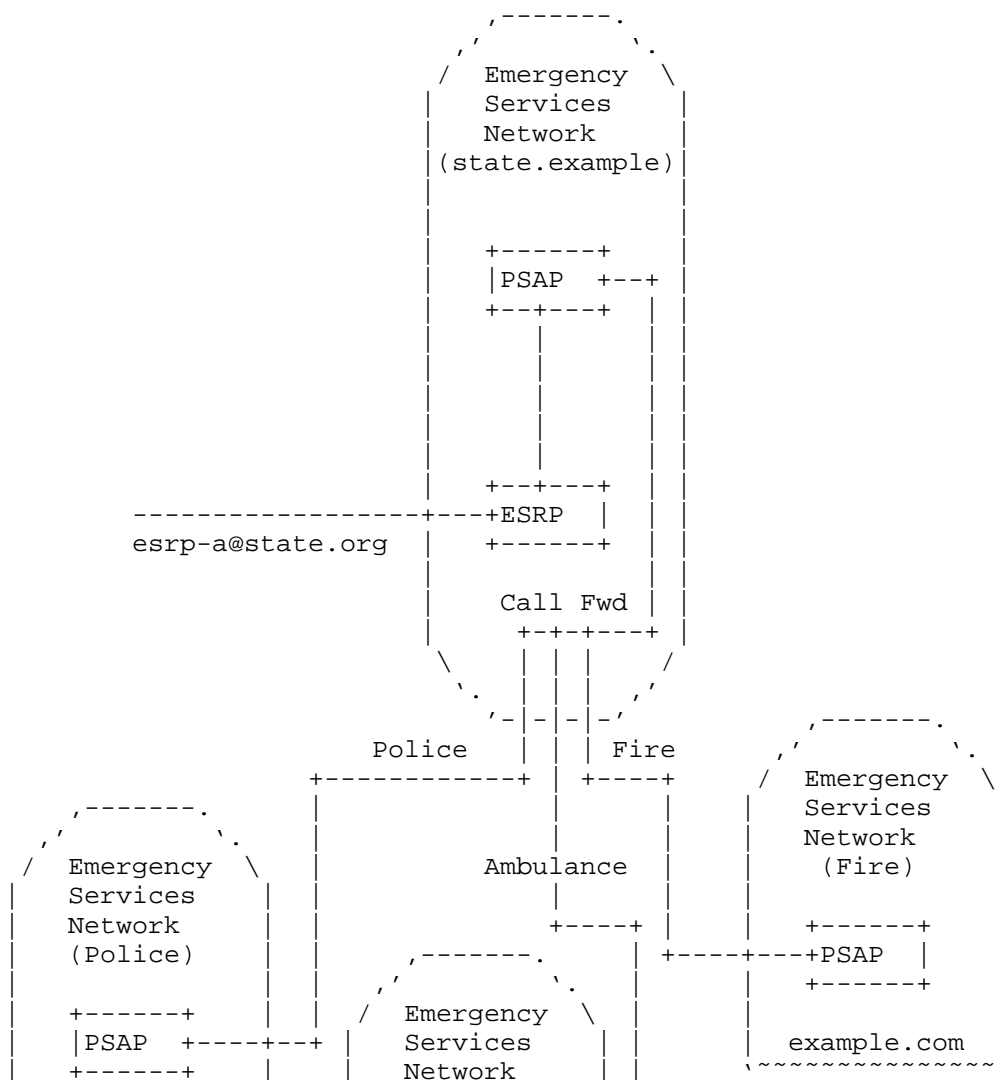
Figure 2: Example for Multi-Stage Routing.

### 3.3. Call Forwarding

Imagine the following case where an emergency call enters an emergency network (`state.example`) via an ESRP but then gets forwarded

to a different emergency services network (in our example to example.net, example.org or example.com). The same considerations apply when the police, fire and ambulance networks are part of the state.example sub-domains (e.g., police.state.example).

Similar to the previous scenario the problem here is with the wrong state information being established during the emergency call setup procedure. A callback would originate in the example.net, example.org or example.com domains whereas the emergency caller's SIP UA or the VoIP outbound proxy has stored state.example.



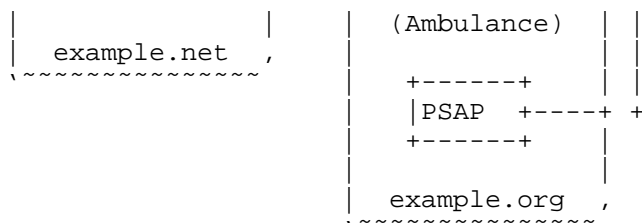


Figure 3: Example for Call Forwarding.

### 3.4. Network-based Service URN Resolution

The IETF emergency services architecture also considers cases where the resolution from the Service URN to the PSAP URI does not only happen at the SIP UA itself but at intermediate SIP entities, such as the user's VoIP provider.

Figure 4 shows this message exchange of the outgoing emergency call and the incoming PSAP graphically. While the state information stored at the VoIP provider is correct the state allocated at the SIP UA is not.

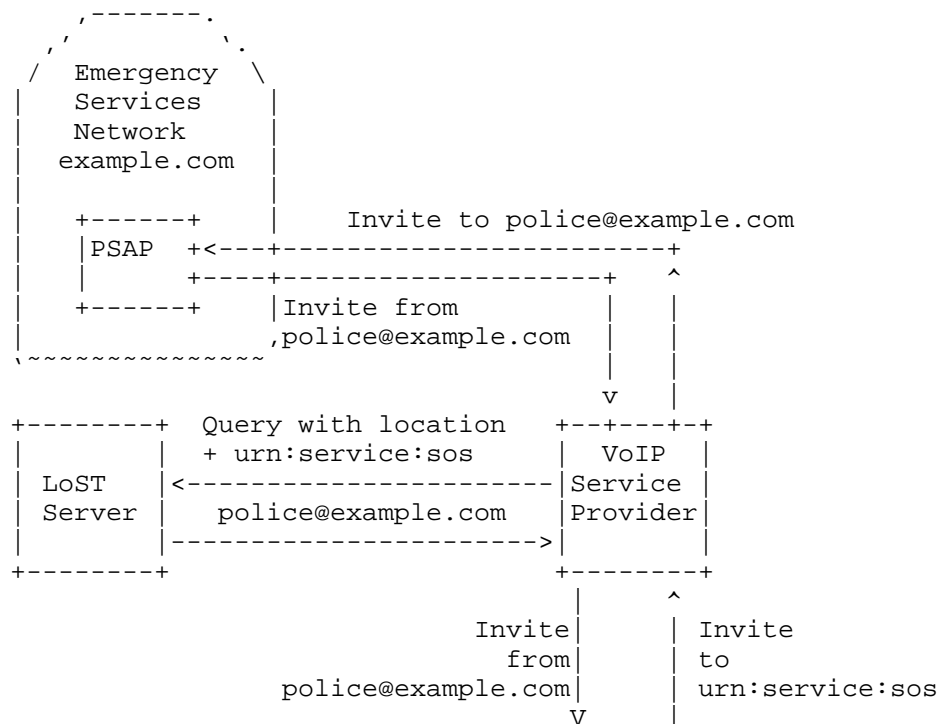




Figure 4: Example for Network-based Service URN Resolution.

### 3.5. PSTN Interworking

In case an emergency call enters the PSTN, as shown in Figure 5, there is no guarantee that the callback some time later leaves the same PSTN/VoIP gateway or that the same end point identifier is used in the forward as well as in the backward direction making it difficult to reliably detect PSAP callbacks.

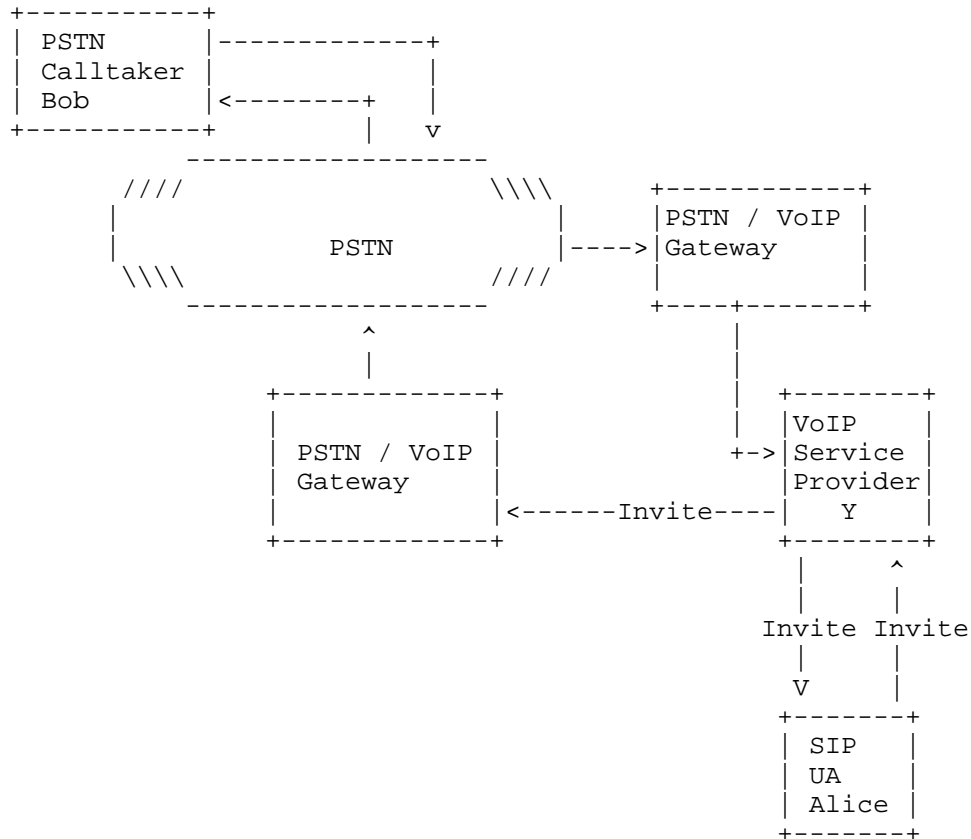


Figure 5: Example for PSTN Interworking.

Note: This scenario is considered outside the scope of this document. The specified solution does not support this use case.

#### 4. SIP PSAP Callback Indicator

##### 4.1. General

This section defines a new header field value, called "psap-callback", for the SIP Priority header field defined in [RFC3261]. The value is used to inform SIP entities that the request is associated with a PSAP callback SIP session.

##### 4.2. Usage

SIP entities that receive the header field value within an initial request for a SIP session can, depending on local policies, apply PSAP callback specific procedures for the session or request.

The PSAP callback specific procedures may be applied by SIP-based network entities and by the callee. The specific procedures taken when receiving such a PSAP callback marked call, such as bypassing services and barring procedures, are outside the scope of this document.

##### 4.3. Syntax

###### 4.3.1. General

This section defines the ABNF for the new SIP Priority header field value "psap-callback".

###### 4.3.2. ABNF

```
priority-value /= "psap-callback"
```

Figure 6: ABNF

#### 5. Security Considerations

##### 5.1. Security Threat

The PSAP callback functionality described in this document allows marked calls to bypass blacklists, ignore call forwarding procedures and other similar features used to raise the attention of emergency callers when attempting to contact them. In the case where the SIP Priority header value, 'psap-callback', is supported by the SIP UA, it would override user interface configurations, such as vibrate-only mode, to alert the caller of the incoming call.

## 5.2. Security Requirements

The security threat discussed in Section 5.1 leads to the requirement to ensure that the mechanisms described in this document can not be used for malicious purposes, including telemarketing.

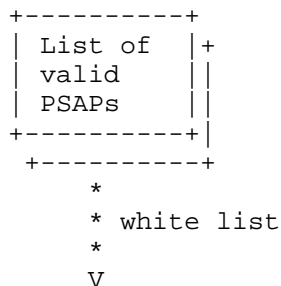
Furthermore, if the newly defined extension is not recognized, not verified adequately, or not obeyed by SIP intermediaries or SIP endpoints then it must not lead to a failure of the call handling procedure. Such call must be treated like a call that does not have any marking attached.

The indicator described in Section 4 can be inserted by any SIP entity, including attackers. So it is critical that the indicator only lead to preferential call treatment in cases where the recipient has some trust in the caller, as described in the next section.

## 5.3. Security Solution

The approach for dealing with implementing the security requirements described in Section 5.2 can be differentiated between the behavior applied by the UA and by SIP proxies. A UA that has made an emergency call MUST keep state information so that it can recognize and accepted a callback from the PSAP if it occurs within a reasonable time after an emergency call was placed, as described in Section 13 of [RFC6443]. Only a timer started at the time when the original emergency call has ended is required; information about the calling party identity is not needed since the callback may use a different calling party identity, as described in Section 3. Since these SIP UA considerations are described already in [RFC6443] as well as in [RFC6881] the rest of this section focuses on the behavior of SIP proxies.

Figure 7 shows the architecture that utilizes the identity of the PSAP to decide whether a preferential treatment of callbacks should be provided. To make this policy decision, the identity of the PSAP (i.e., calling party identity) is compared with a PSAPs white list.



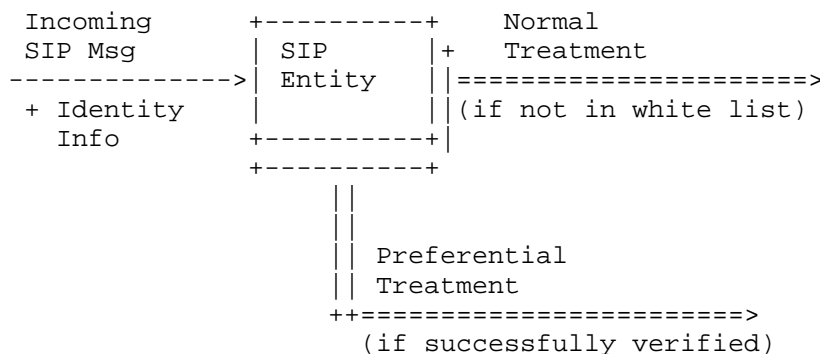


Figure 7: Identity-based Authorization

The identity assurance in SIP can come in different forms, namely via the SIP Identity [RFC4474] or the P-Asserted-Identity [RFC3325] mechanisms. The former technique relies on a cryptographic assurance and the latter on a chain of trust. Also the usage of TLS between neighboring SIP entities may provide useful identity information. At the time of writing these identity technologies are being revised in the Secure Telephone Identity Revisited (stir) working group [STIR] to offer better support for legacy technologies interworking and SIP intermediaries that modify the content of various SIP headers and the body. Once the work on these specifications has been completed they will offer a stronger calling party identity mechanism that limits or prevents identity spoofing.

An important aspect from a security point of view is the relationship between the emergency services network (containing the PSAPs) and the VoIP provider (assuming that the emergency call travels via the VoIP provider and not directly between the SIP UA and the PSAP).

The establishment of a white list with PSAP identities may be operationally complex and dependent on the relationship between the emergency services operator and the VoIP provider. When there is a relationship between the VoIP provider and the PSAP operator, for example when they are both operating in the same geographical region, then populating the white list is fairly simple and consequently the identification of a PSAP callback is less problematic compared to the case where the two entities have never interacted with each other before. In the end, the VoIP provider has to verify whether the marked callback message indeed came from a legitimate source.

VoIP providers **MUST** only give PSAP callbacks preferential treatment when the calling party identity of the PSAP was successfully matched against entries in the white list. If it cannot be verified (because there was no match), then the VoIP provider **MUST** remove the PSAP

callback marking. Thereby, the callback is degenerated to a normal call. As a second step, SIP UAs MUST maintain a timer that is started with the original emergency call and this timer expires within a reasonable amount of time, such as 30 minutes per [RFC6881]. Such a timer also ensures that VoIP providers cannot misuse the PSAP callback mechanism, for example to ensure that their support calls reaches their customers.

Finally, a PSAP callback MUST use the same media as the original emergency call. For example, when an initial emergency call established a real-time text communication session then the PSAP callback must also attempt to establish a real-time communication interaction. The reason for this is two-fold. First, the person seeking for help may have disabilities that prevent them from using certain media and hence using the same media for the callback avoids unpleasant surprises and delays. Second, the emergency caller may have intentionally chosen a certain media and does not prefer to communicate in a different way. For example, it would be unfortunate if a hostage tries to seek for help using instant messaging to avoid any noise when subsequently the ring-tone triggered by a PSAP callback using a voice call gets the attention of the hostage-taker. User interface designs need to cater to such situations.

## 6. IANA Considerations

This document adds the "psap-callback" value to the SIP Priority header IANA registry allocated by [RFC6878]. The semantic of the newly defined "psap-callback" value is defined in Section 4.

## 7. Acknowledgements

We would like to thank the following persons for their feedback: Paul Kyzivat, Martin Thomson, Robert Sparks, Keith Drage, Cullen Jennings, Brian Rosen, Martin Dolly, Bernard Aboba, Andrew Allen, Atle Monrad, John-Luc Bakker, John Elwell, Geoff Thompson, Dan Romascanu, James Polk, John Medland, Hadriel Kaplan, Kenneth Carlberg, Timothy Dwight, Janet Gunn

We would like to thank the ECRIT working group chairs, Marc Linsner and Roger Marshall, for their support. Roger Marshall was the document shepherd for this document. Vijay Gurbani provided the general area review.

During IESG review the document received good feedback from Barry Leiba, Spencer Dawkins, Richard Barnes, Joel Jaeggli, Stephen Farrell, and Benoit Claise.



## 8. References

### 8.1. Normative References

- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.
- [RFC5627] Rosenberg, J., "Obtaining and Using Globally Routable User Agent URIs (GRUUs) in the Session Initiation Protocol (SIP)", RFC 5627, October 2009.
- [RFC6878] Roach, A., "IANA Registry for the Session Initiation Protocol (SIP) "Priority" Header Field", RFC 6878, March 2013.

### 8.2. Informative References

- [RFC3325] Jennings, C., Peterson, J., and M. Watson, "Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks", RFC 3325, November 2002.
- [RFC4474] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, August 2006.
- [RFC5012] Schulzrinne, H. and R. Marshall, "Requirements for Emergency Context Resolution with Internet Technologies", RFC 5012, January 2008.
- [RFC5222] Hardie, T., Newton, A., Schulzrinne, H., and H. Tschofenig, "LoST: A Location-to-Service Translation Protocol", RFC 5222, August 2008.
- [RFC6443] Rosen, B., Schulzrinne, H., Polk, J., and A. Newton, "Framework for Emergency Calling Using Internet Multimedia", RFC 6443, December 2011.
- [RFC6881] Rosen, B. and J. Polk, "Best Current Practice for Communications Services in Support of Emergency Calling", BCP 181, RFC 6881, March 2013.
- [STIR] IETF, "Secure Telephone Identity Revisited (stir) Working Group", URL: <http://datatracker.ietf.org/wg/stir/charter/>, Oct 2013.

Authors' Addresses

Henning Schulzrinne  
Columbia University  
Department of Computer Science  
450 Computer Science Building  
New York, NY 10027  
US

Phone: +1 212 939 7004  
EMail: [hgs+ecrit@cs.columbia.edu](mailto:hgs+ecrit@cs.columbia.edu)  
URI: <http://www.cs.columbia.edu>

Hannes Tschofenig  
Nokia Solutions and Networks  
Linnoitustie 6  
Espoo 02600  
Finland

Phone: +358 (50) 4871445  
EMail: [Hannes.Tschofenig@gmx.net](mailto:Hannes.Tschofenig@gmx.net)  
URI: <http://www.tschofenig.priv.at>

Christer Holmberg  
Ericsson  
Hirsalantie 11  
Jorvas 02420  
Finland

EMail: [christer.holmberg@ericsson.com](mailto:christer.holmberg@ericsson.com)

Milan Patel  
InterDigital Communications

EMail: [Milan.Patel@interdigital.com](mailto:Milan.Patel@interdigital.com)

This Internet-Draft, draft-ietf-ecrit-rough-loc-04.txt, has expired, and has been deleted from the Internet-Drafts directory. An Internet-Draft expires 185 days from the date that it is posted unless it is replaced by an updated version, or the Secretariat has been notified that the document is under official review by the IESG or has been passed to the RFC Editor for review and/or publication as an RFC. This Internet-Draft was not published as an RFC.

Internet-Drafts are not archival documents, and copies of Internet-Drafts that have been deleted from the directory are not available. The Secretariat does not have any information regarding the future plans of the authors or working group, if applicable, with respect to this deleted Internet-Draft. For more information, or to request a copy of the document, please contact the authors directly.

Draft Authors:

Richard Barnes<rbarnes@bbn.com>

Matt Lepinski<mlepinski@bbn.com>

ECRIT  
Internet-Draft  
Intended status: Informational  
Expires: January 13, 2014

B. Rosen  
NeuStar, Inc.  
H. Tschofenig  
Nokia Siemens Networks  
R. Gellens  
QUALCOMM Incorporated  
July 14, 2013

Internet Protocol-based In-Vehicle Emergency Call  
draft-rosen-ecrit-ecall-10.txt

Abstract

This document describes how to re-use the emergency services mechanisms specified for the Session Initiation Protocol (SIP) to accomplishing emergency calling support in vehicles. Profiling and simplifications are possible due to the nature of the functionality that is going to be provided in vehicles with the usage of Global Positioning System (GPS).

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 13, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction . . . . .	2
1.1.	Overview of Current Deployment Models . . . . .	3
1.2.	Migration to IP-based Models . . . . .	4
2.	Terminology . . . . .	5
3.	Profile . . . . .	5
4.	Example . . . . .	6
5.	Security Considerations . . . . .	9
6.	IANA Considerations . . . . .	9
6.1.	Service URN Registration . . . . .	9
6.2.	MIME Content-type Registration for 'application/emergencyCall.VEDS+xml' . . . . .	9
6.3.	Registration of the 'VEDS' entry in the Emergency Call Additional Data registry . . . . .	10
7.	Contributors . . . . .	10
8.	Acknowledgements . . . . .	11
9.	References . . . . .	11
9.1.	Normative References . . . . .	11
9.2.	Informative references . . . . .	11
	Appendix A. Matching Functionality with eCall Minimum Set of Data (MSD) . . . . .	12
	Authors' Addresses . . . . .	13

## 1. Introduction

Emergency calls made from vehicles can assist with the objective of significantly reducing road deaths and injuries. Unfortunately, drivers often have a poor location-awareness, especially on urban roads (also during night) and abroad. In the most crucial cases, the victim(s) may not be able to call because they have been injured or trapped.

In Europe the European Commission has launched the 'eCall' initiative that may best be described as a user-initiated or automatically triggered system to provide notifications to Public Safety Answering Points (PSAPs), by means of cellular communications, that a vehicle has crashed, and to provide geodetic location information and where possible a voice channel to the PSAP.

The general term for such systems is Automatic Crash Notification (ACN). ACN systems transmit some amount of data specific to the incident, referred to generally as "crash data." While different systems transmit different amounts of crash data, standardized formats, structures, and mechanisms are needed to provide interoperability among systems and PSAPs.

This document describes how existing IETF mechanisms are used to provide the realization of next-generation ACN in general, including European eCall.

This document registers the 'application/emergencyCall.VEDS+xml' MIME content-type, and registers the 'VEDS' entry in the Emergency Call Additional Data registry.

The Vehicle Emergency Data Set (VEDS) is an XML structure defined by the Association of Public-Safety Communications Officials (APCO) and the National Emergency Number Association (NENA). The 'application/emergencyCall.VEDS+xml' MIME content-type is used to identify it. The 'VEDS' entry in the Emergency Call Additional Data registry is used to construct a 'purpose' parameter value for conveying VEDS data in a Call-Info header.

Circuit-switched eCall systems transmit crash data as a defined set, the Minimum Set of Data (MSD) [eCall-MSD]. The MSD for circuit-switched eCall is a binary format defined by CEN, the European Committee for Standardization. It is expected that CEN will choose to define the XML schema for the eCall MSD for use in next-generation systems. Once this done, a MIME content-type (e.g., 'application/emergencyCall.eCall.MSD+xml') and Emergency Call Additional Data entry (e.g., 'eCall.MSD') need to be registered for the MSD. Note that Appendix A explains how the functionality available in IETF specifications maps to the functionality required for the MSD of the mobile circuit switched voice solution.

CEN and/or other entities may define additional sets of data in the same manor: a standardized format, such as XML, is defined, and a MIME content-type and Emergency Call Additional Data entry registered.

An In-Vehicle System (IVS) transmits crash data by encoding it in one of the standardized and registered formats (such as VEDS or eCall.MSD) and attaching it to an INVITE as a data block. The block is identified by its MIME content-type, and pointed to by a CID URL in a Call-Info header with a 'purpose' parameter value corresponding to the block.

The mechanisms described here can be used to deploy ACN systems in general including eCall by providing for emergency calls that are identifiable as ACN calls or specifically eCall calls and that carry one or more defined crash data objects.

### 1.1. Overview of Current Deployment Models

Current (circuit-switched or legacy) systems for placing emergency calls from vehicles, including automatic crash notification system, generally use one of three architectural models: Telematics Service Provider (TSP), direct, and paired handset. These three models are illustrated below.

In the TSP model the IVS transmits crash data to the TSP using proprietary means. The TSP operator bridges in the PSAP and communicates location, crash, and other data to the call taker verbally (there is a three-way voice call between the vehicle, the TSP, and the PSAP).

```

    ///----\\  proprietary  +-----+    911 trunk    +-----+
    ||| IVS |||----->+ TSP  +----->+ PSAP |
    \\----///  crash data   +-----+                +-----+

```

In the paired model the IVS uses a Bluetooth link to a previously-paired handset to establish an emergency call with the PSAP and then communicates location data to the PSAP via text-to-speech; crash data is not conveyed.

```

    ///----\\      ++
    ||| IVS |||---->|  911 voice call via handset    +-----+
    \\----///      ++  location via text-to-speech    +-----+

```

In the direct model the IVS communicates crash data to the PSAP via the eCall in-band modem (in the voice call).

```

    ///----\\      112/911 voice call via IVS    +-----+
    ||| IVS |||----->+ PSAP |
    \\----///  crash data via eCall in-band modem  +-----+

```

## 1.2. Migration to IP-based Models

The migration to next-generation (all-IP) would then look like as follows.

In the TSP model The IVS transmits crash data to the TSP using either proprietary or standard means. The TSP bridges in the PSAP and transmits crash and other data to the PSAP using IETF specifications. There is a three-way call between the vehicle, the TSP, and the PSAP.

```

    ///----\\      proprietary
    ||| IVS |||----->+ TSP  +-----+    standard    +-----+
    \\----///  crash data   +-----+ crash + other data +-----+

```

In the paired model, the IVS uses a Bluetooth link to a previously-paired handset to establish an emergency call with the PSAP; it is not clear what facilities are or will be available for transmitting crash data.

```

    ///----\\      ++
    ||| IVS |||---->|  IP-based Emergency Call    +-----+
    \\----///      ++                                +-----+

```

In the direct model the IVS communicates crash data to PSAP using Internet protocols.

```

///----\\\           NG1-1-2/NG9-1-1 call           +-----+
   IVS  ----->+ PSAP |
\\----///           crash data                       +-----+

```

This document is focused on the interface to the PSAP, that is, how an emergency call (including location and crash data) is setup and data is transmitted to the PSAP using existing IETF specifications. The goal is to re-use existing specifications rather than to invent new. For the direct model (such as the European eCall), this is the end-to-end description. For the TSP model, this describes the right-hand side, leaving the left-hand side up to the entities involved (e.g., IVS and TSP vendors) who are then free to use the same mechanism as for the right-hand side or not.

## 2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

This document re-uses terminology defined in Section 3 of [RFC5012].

Additionally, we use the following abbreviations:

IVS: In-Vehicle System

TSP: Telematics Service Provider

MSD: Minimum Set of Data

VEDS: Vehicle Emergency Data Set

NENA: National Emergency Number Association

APCO: Association of Public-Safety Communications Officials

CEN: European Committee for Standardization

ESInet: Emergency Services IP network

## 3. Profile

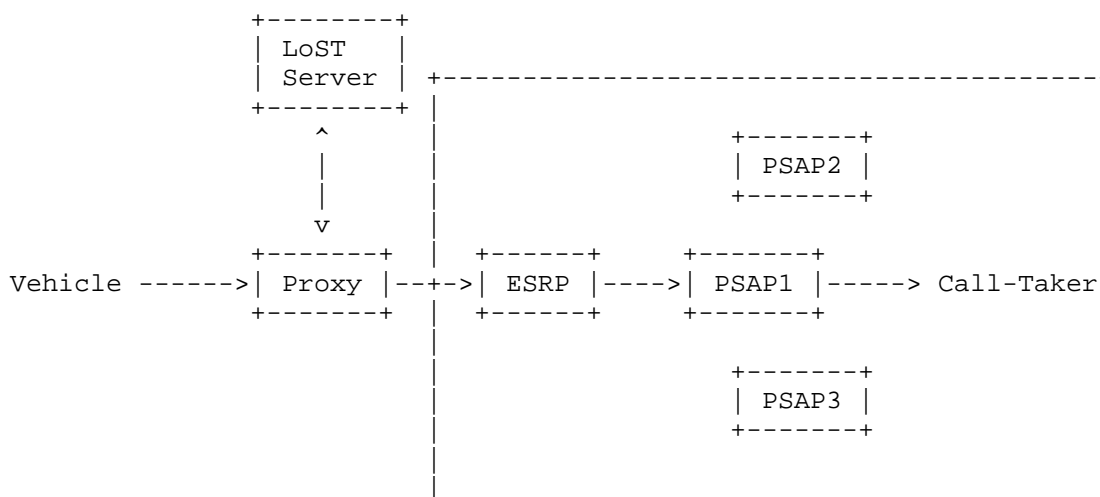


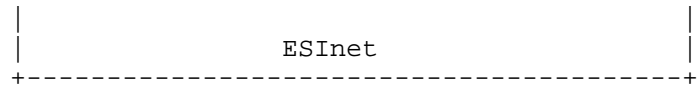
In the context of emergency calls placed from a vehicle it is assumed that the car is equipped with a built-in GPS receiver. For this reason only geodetic location information will be sent within an emergency call. The following location shapes MUST be implemented: 2d and 3d Point (see Section 5.2.1 of [RFC5491]), Circle (see Section 5.2.3 of [RFC5491]), and Ellipsoid (see Section 5.2.7 of [RFC5491]). The coordinate reference systems (CRS) specified in [RFC5491] are also mandatory for this document. The <direction> element, as defined in [RFC5962] which indicates the direction of travel of the vehicle, is important for dispatch and hence it MUST be included in the PIDF-LO. The <heading> element specified in [RFC5962] MUST be implemented and MAY be included.

This specification also inherits the ability to utilize test call functionality from Section 15 of [RFC6881].

#### 4. Example

Figure 7 shows an emergency call placed from a vehicle whereby location information is directly attached to the SIP INVITE message itself. The call uses the request URI 'urn:service:sos.ecall.automatic' service URN and is recognized as an emergency call because the request URI starts with 'urn:service:sos'. The VoIP provider routes the call to an Emergency services IP Network (ESInet), as for any emergency call. The ESInet routes the call to an appropriate PSAP using location information and the fact that that it is an eCall carrying crash data. (In deployments where there is no ESInet, the VoIP provider may route directly to an appropriate PSAP.) The emergency call continues towards the PSAP and in this example it hits the ESRP, as the entry point to the ESInet. Finally, the emergency call will be received by a call taker and first responders will be dispatched.





The example, shown in Figure 8, illustrates a SIP emergency call eCall INVITE that is being conveyed with location information encoded in a PIDF-LO and VEDS data.

```
INVITE urn:service:sos.ecall.automatic SIP/2.0
To: urn:service:sos.ecall.automatic
From: <sip:+13145551111@example.com>;tag=9fxced76s1
Call-ID: 3848276298220188511@atlanta.example.com
Geolocation: <cid:target123@example.com>
Geolocation-Routing: no
Call-Info: cid:1234567890@atlanta.example.com;
           purpose=emergencyCallData.VEDS
Accept: application/sdp, application/pidf+xml
CSeq: 31862 INVITE
Content-Type: multipart/mixed; boundary=boundary1
Content-Length: ...
```

```
--boundary1
```

```
Content-Type: application/sdp
```

```
...Session Description Protocol (SDP) goes here
```

```
--boundary1
```

```
Content-Type: application/pidf+xml
Content-ID: <target123@atlanta.example.com>
<?xml version="1.0" encoding="UTF-8"?>
<presence
  xmlns="urn:ietf:params:xml:ns:pidf"
  xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
  xmlns:gp="urn:ietf:params:xml:ns:pidf:geopriv10"
  xmlns:dyn="urn:ietf:params:xml:ns:pidf:geopriv10:dynamic"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:gs="http://www.opengis.net/pidflo/1.0"
  entity="sip:+13145551111@example.com">
  <dm:device id="123">
    <gp:geopriv>
      <gp:location-info>
        <gml:Point srsName="urn:ogc:def:crs:EPSG::4326">
          <gml:pos>-34.407 150.883</gml:pos>
        </gml:Point>
        <dyn:Dynamic>
          <dyn:heading>278</dyn:heading>
          <dyn:direction><dyn:direction>
        </dyn:Dynamic>
      </gp:location-info>
      <gp:usage-rules/>
      <method>gps</method>
    </gp:geopriv>
    <timestamp>2012-04-5T10:18:29Z</timestamp>
    <dm:deviceID>1M8GDM9A_KP042788</dm:deviceID>
  </dm:device>
</presence>

--boundary1
```

Content-Type: application/emergencyCall.VEDS+xml

Content-ID: 1234567890@atlanta.example.com

...eCall VEDS data object goes here

--boundary1--

## 5. Security Considerations

This document does not raise security considerations beyond those described in [RFC5069]. As with emergency service systems with end host provided location information there is the possibility that that location is incorrect, either intentionally (in case of an a denial of service attack against the emergency services infrastructure) or due to a malfunctioning devices. The reader is referred to [I-D.ietf-ecrit-trustworthy-location] for a discussion of some of these vulnerabilities.

## 6. IANA Considerations

### 6.1. Service URN Registration

IANA is requested to register the URN 'urn:service:sos.ecall' under the sub-services 'sos' registry defined in Section 4.2 of [RFC5031].

This service identifier reaches a public safety answering point (PSAP), which in turn dispatches aid appropriate to the emergency related to accidents of vehicles. Two sub-services are registered as well, namely

urn:service:sos.ecall.manual

This service URN indicates that an eCall had been triggered based on the manual interaction of the driver or a passenger.

urn:service:sos.ecall.automatic

This service URN indicates that an eCall had been triggered automatically, for example, due to a crash. No human involvement was detected.

### 6.2. MIME Content-type Registration for 'application/emergencyCall.VEDS+xml'

This specification requests the registration of a new MIME type according to the procedures of RFC 4288 [RFC4288] and guidelines in RFC 3023 [RFC3023].

MIME media type name: application

MIME subtype name: emergencyCall.VEDS+xml

Mandatory parameters: none

Optional parameters: charset

Indicates the character encoding of enclosed XML.

Encoding considerations: Uses XML, which can employ 8-bit characters, depending on the character encoding used. See Section 3.2 of RFC 3023 [RFC3023].

Security considerations: This content type is designed to carry vehicle crash data during an emergency call. This data may contain personal information including vehicle VIN, location, direction, etc. appropriate precautions need to be taken to limit unauthorized access, inappropriate disclosure to third parties, and eavesdropping of this information. Please refer to Section 7 and Section 8 of [I-D.ietf-ecrit-additional-data] for more information.

Interoperability considerations: None

Published specification: [TBD: This specification]

Applications which use this media type: Emergency Services

Additional information: None

Magic Number: None

File Extension: .xml

Macintosh file type code: 'TEXT'

Person and email address for further information: Hannes Tschofenig, Hannes.Tschofenig@gmx.net

Intended usage: LIMITED USE

Author: This specification is a work item of the IETF ECrit working group, with mailing list address <ecrit@ietf.org>.

Change controller: The IESG <ietf@ietf.org>

### 6.3. Registration of the 'VEDS' entry in the Emergency Call Additional Data registry

This specification requests IANA to add the 'VEDS' entry to the Emergency Call Additional Data registry, with a reference to this document. The Emergency Call Additional Data registry has been established by [I-D.ietf-ecrit-additional-data].

## 7. Contributors

We would like to thank Ulrich Dietz for his help with earlier versions of the document.

## 8. Acknowledgements

We would like to thank Michael Montag, Arnoud van Wijk, Ban Al-Bakri, and Gunnar Hellstroem for their feedback.

## 9. References

### 9.1. Normative References

- [10] Schulzrinne, H., "A Uniform Resource Name (URN) for Emergency and Other Well-Known Services", RFC 5031, January 2008.
- [11] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [12] Peterson, J., "A Presence-based GEOPRIV Location Object Format", RFC 4119, December 2005.
- [13] Winterbottom, J., Thomson, M. and H. Tschofenig, "GEOPRIV Presence Information Data Format Location Object (PIDF-LO) Usage Clarification, Considerations, and Recommendations", RFC 5491, March 2009.
- [14] Rosen, B. and J. Polk, "Best Current Practice for Communications Services in Support of Emergency Calling", BCP 181, RFC 6881, March 2013.
- [15] Polk, J., Rosen, B. and J. Peterson, "Location Conveyance for the Session Initiation Protocol", RFC 6442, December 2011.
- [16] Schulzrinne, H., Singh, V., Tschofenig, H. and M. Thomson, "Dynamic Extensions to the Presence Information Data Format Location Object (PIDF-LO)", RFC 5962, September 2010.
- [17] Murata, M., St. Laurent, S. and D. Kohn, "XML Media Types", RFC 3023, January 2001.
- [18] Freed, N. and J. Klensin, "Media Type Specifications and Registration Procedures", RFC 4288, December 2005.
- [19] Rosen, B., Tschofenig, H., Marshall, R. and R. Randy, "Additional Data related to an Emergency Call", Internet-Draft draft-ietf-ecrit-additional-data-09, May 2013.

### 9.2. Informative references

- [1] Schulzrinne, H. and R. Marshall, "Requirements for Emergency Context Resolution with Internet Technologies", RFC 5012, January 2008.

- [2] Taylor, T., Tschofenig, H., Schulzrinne, H. and M. Shanmugam, "Security Threats and Requirements for Emergency Call Marking and Mapping", RFC 5069, January 2008.
- [3] Tschofenig, H., Schulzrinne, H. and B. Aboba, "Trustworthy Location", Internet-Draft draft-ietf-ecrit-trustworthy-location-05, March 2013.
- [4] Schulzrinne, H., "Timed Presence Extensions to the Presence Information Data Format (PIDF) to Indicate Status Information for Past and Future Time Intervals", RFC 4481, July 2006.
- [5] CEN, , "Intelligent transport systems - eSafety - eCall minimum set of data (MSD), EN 15722", June 2011.

#### Appendix A. Matching Functionality with eCall Minimum Set of Data (MSD)

[eCall-MSD] outlines a number of data elements that are transmitted in an emergency call triggered by a vehicle. Note that the work on eCall for mobile circuit switched voice is constrained in a number of ways since legacy eCall uses an inband voice modem for backwards compatibility with the already deployed cellular infrastructure to transmit data from a vehicle to a PSAP. Since the functionality in this document is based on the Session Initiation Protocol (SIP) these limitations do not exist. As such, it is not useful to transmit the MSD inband in the voice channel but to rather use the SIP mechanisms standardized for emergency call handling. Any voice, video, or real-text communication will be negotiated using the Session Description Protocol (SDP), as shown in Figure 8, and the actual media stream will then take place in RTP packets. For transmitting location information an XML-based data structure had been defined, the so-called Presence Information Data Format Location Object (PIDF-LO).

The following list compares the eCall minimum set of data with the functionality provided in this document.

Version of the MSD Format: Conveying information in a SIP-based emergency call is accomplished by using XML payloads and XML provides namespace declarations that allow a recipient of that information to distinguish different versions and additional extensions. For example, if additional data about a vehicle is defined and can be transmitted by vehicle then a respective extension can be defined for use inside a previously-defined XML structure. One or more top-level structures can be transmitted using the mechanism defined in [I-D.ietf-ecrit-additional-data]. Selecting the appropriate extension point depends on the type of extension envisioned.

Message Identifier: Every SIP INVITE message contains a Call-ID, which is a globally unique identifier for this call.

Test Call Indication A service URN starting with "test." indicates a request for an automated test. For example, "urn:service:test.sos.ecall.automatic" indicates such a test feature. This functionality is defined in [RFC6881].

Automatic Activation Indication: This document registers new service URNs, which allow the differentiation between manually and automatically triggered emergency calls. The two service URNs are: urn:service:sos.ecall.automatic and urn:service:sos.ecall.manual

Vehicle Identification: The PIDF data structure contains a deviceID field that holds the Vehicle Identification Number (VIN).

Timestamp of Incident Event: The PIDF-LO element contains the timestamp when the PIDF-LO was created, which is at the time of the incident.

Vehicle Location: The location of the vehicle is conveyed using the PIDF location object, as described in Section 3.

Vehicle Direction: The direction of the vehicle is part of location information, as described in Section 3.

Recent Vehicle Location: With this optional functionality multiple location objects may be required to be transported simultaneously. This can be achieved using <timed-presence>, defined in RFC 4481 [RFC4481].

Additional Data: [I-D.ietf-ecrit-additional-data] provides the ability to carry additional data for an emergency call.

While most fields have an equivalent already in the corresponding SIP emergency signaling payloads there are currently no fields defined in [I-D.ietf-ecrit-additional-data] that allow information about the "Vehicle Type Encoding", "Number of Passengers", and "Vehicle Propulsion Storage type" to be conveyed. Extensions for those fields will have to be defined.

#### Authors' Addresses

Brian Rosen  
NeuStar, Inc.  
470 Conrad Dr  
Mars, PA 16046  
US

Email: br@brianrosen.net



Hannes Tschofenig  
Nokia Siemens Networks  
Linnoitustie 6  
Espoo, 02600  
Finland

Phone: +358 (50) 4871445  
Email: Hannes.Tschofenig@gmx.net  
URI: <http://www.tschofenig.priv.at>

Randall Gellens  
QUALCOMM Incorporated  
5775 Morehouse Drive  
San Diego, 92651  
US

Email: [rg+ietf@qualcomm.com](mailto:rg+ietf@qualcomm.com)

ECRIT  
Internet-Draft  
Updates: RFC6881 (if approved)  
Intended status: Standards Track  
Expires: November 30, 2014

J. Winterbottom  
Winterb Consulting Services  
H. Tschofenig

L. Liess  
Deutsche Telekom  
May 29, 2014

A Routing Request Extension for the HELD Protocol  
draft-winterbottom-ecrit-priv-loc-04.txt

## Abstract

In many circumstances public LoST servers or a distributed network of forest guides linking public LoST servers is not available. In such environments the general ECRIT calling models breakdown. However, location servers operating in these areas are often privy to the necessary information to reach emergency and other services. This document describes a solution where by the routing information may be obtained from a location server using a simple extension to the HELD protocol.

## Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on November 30, 2014.

## Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of

publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	3
3. Motivation . . . . .	4
4. Mechanism . . . . .	6
5. HELD Schema Extension . . . . .	8
6. Examples . . . . .	9
7. Privacy Considerations . . . . .	10
8. Security Considerations . . . . .	10
9. IANA Considerations . . . . .	11
9.1. URN sub-namespace registration for 'urn:ietf:params:xml:ns:geopriv:held:ri' . . . . .	11
9.2. XML Schema Registration . . . . .	11
10. Acknowledgements . . . . .	12
11. References . . . . .	12
11.1. Normative References . . . . .	12
11.2. Informative References . . . . .	12
Authors' Addresses . . . . .	13

## 1. Introduction

In many circumstances public LoST [RFC5222] servers or a distributed network of forest guides linking public LoST servers is not available. In such environments the general ECRIT calling models breakdown. Location servers operating in these areas are often privy to the necessary information to reach emergency and other services. This document describes how adding an extension to the HELD protocol [RFC5985] can be used to extract this information for a location information server in the absence of a LoST server or network of forest guides.

## 2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

The terms LIS, ESRP, VSP and PSAP are used as defined in [RFC6443].

The term "Access Network Provider" is used as defined in [RFC5687] and encompasses both the Internet Access Provider (IAP) and Internet Service Provider (ISP).

### 3. Motivation

The Internet emergency calling architecture specified in [RFC6881] describes two main models for emergency call processing. The first is a device-centric model, where a device obtains location information using a location configuration protocol, such a HELD [RFC5985], and then proceeds to determine the address of the next hop closer to the local PSAP using LoST [RFC5222]. Figure 1 shows this model in a simplified form.

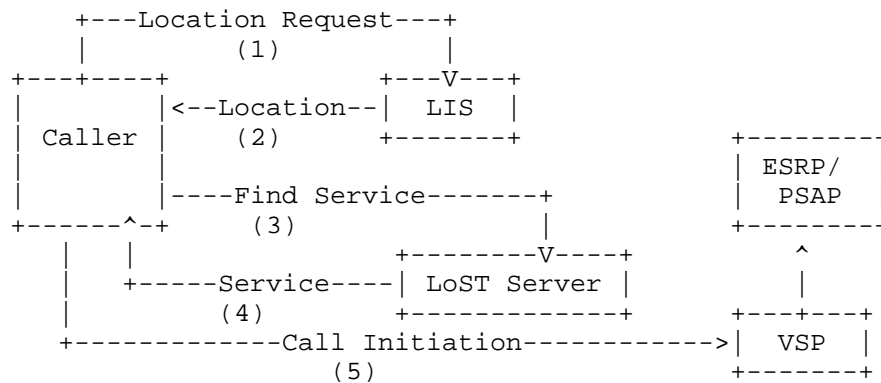


Figure 1: Device-Centric Emergency Services Model

The second approach is a softswitch-centric model, where a device initiates an emergency call and the serving softswitch detects that the call is an emergency and initiates retrieving the caller's location from a Location Information Server (LIS) using HELD [RFC5985] with identity extensions [RFC6155] [RFC6915] and then determining the route to the local PSAP using LoST [RFC5222]. Figure 2 shows the high-level protocol interactions.

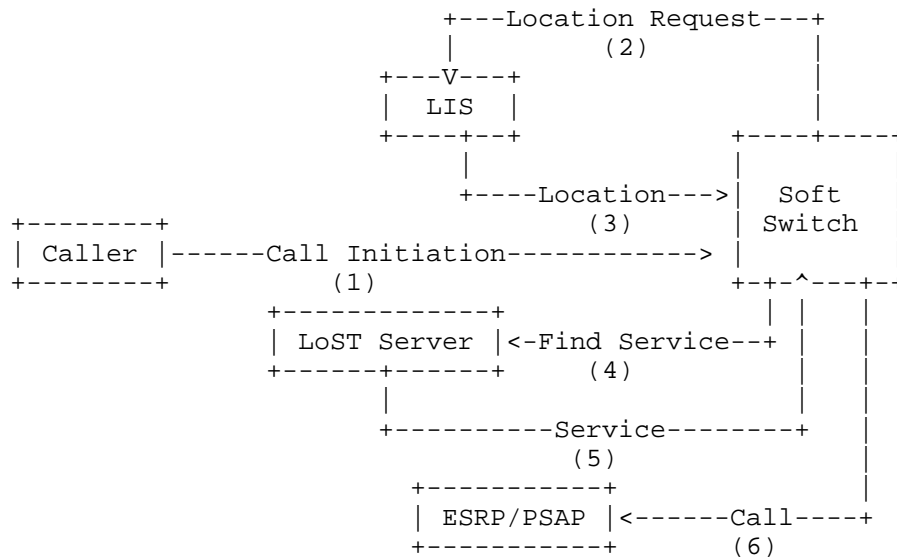


Figure 2: Softswitch-Centric Calling Model

In the softswitch-centric model when a VSP receives an emergency call it performs two tasks. The first task is to determine the correct LIS to ask for location information, this is done using a combination of reverse DNS lookup described in [RFC7216] to acquire the serving domain name and then using [RFC5986] to determine the LIS URI. Once the location is obtained from the LIS, the VSP determines the LoST server associated with the domain serving the caller and queries it for the correct PSAP address.

LoST server discovery is a domain based activity, similar to the LIS discovery technique. However, unlike the LIS that is a domain bound service, a LoST server is a geographically bound service. This means that for a domain that spans multiple geographic regions the LoST server determined may not be able to provide a route to the necessary PSAP. When this occurs, the contacted LoST server invokes the help of other LoST servers and this requires the deployment of forest guides.

At the time of writing, several countries have expressed their reluctance to deploy public LoST servers. In countries amenable to use of LoST and forest guides no public forest guides have been deployed. There appears little interest from the public sector in establishing a global forest guide network. These issues pose threats to both the device-centric and the softswitch-centric calling approaches in terms of them operating everywhere.

The device-centric and softswitch-centric calling models both involve the notion of a LIS bound to the serving access network. In many cases the LIS already knows the destination PSAP address for any given location. In [RFC6881] for example, the LIS validates all civic locations using a location validation procedure. This procedure is the same as a routing request and so the LIS has the resulting the PSAP routing information. In other cases, the LIS knows the correct PSAP for a given location at provisioning time, or the access network might always route to the same emergency provider. Irrespective of the way in which the LIS learns the PSAP address for a location, the LIS will, in a great many cases, have this information.

This document specifies an extension to the HELD protocol so that emergency routing information can be requested from the LIS at the same time that location information is requested. The document updates [RFC6881] by requiring devices and softswitches that understand this specification to always request routing information to avoid the risk of query failure where no LoST server or forest guide network is deployed.

#### 4. Mechanism

The mechanism consists of adding an element to the HELD locationRequest and an element to the locationResponse. The request element indicates that the requestor wants the LIS to provide routing information for the location where the device is. If the LIS understands the routing request and has routing information accessible it provides the information in a routingInformation element included in the locationResponse. How the LIS obtains this information is left to implementation, one possible option is that the LIS acquires it from a LoST server, other possibilities are described in Section 3.

A LIS that does not understand the routing request element ignores it and returns location as normal.

A LIS that does understand the routing request element but can't obtain routing information returns location as normal.

The routing information in the location response consists of one or more service elements which is identified by a service name. The service name is a URI and might contain a general emergency service urn such as urn:service:sos or might contain a specific service urn. For each service name a list of one or more service destinations is provided. Each destination is expressed as a URI and each URI scheme should only appear once in this list. The routing information is

intended to be used at the time it is received. To avoid any risks of using stale routing information the value should not be cached by the receiving entity.

Reusing the mapping element from the LoST findServiceResponse message to provide the routing information was considered. However, this would have meant that several of the mandatory components in the mapping element would have had to contain ambiguous or misleading values. Specifically, the "source" attribute is required to contain a LoST application unique string for the authoritative server. However, in the situations described in this specification there may not be an authoritative LoST server, so any value put into this attribute would be misleading. In addition to this, routing information received in the manner described in this specification should not be cached by the receiver, so detailing when the routing information expires or was last updated is irrelevant.



## 5. HELD Schema Extension

This section describes the schema extension to HELD.

```
<?xml version="1.0"?>
<xs:schema
  targetNamespace="urn:ietf:params:xml:ns:geopriv:held:ri"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:eri="urn:ietf:params:xml:ns:geopriv:held:ri"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xs:element name="requestRoutingInformation">
    <xs:complexType name="empty"/>
  </xs:element>

  <xs:complexType name="service">
    <xs:complexContent>
      <xs:restriction base="xs:anyType">
        <xs:sequence>
          <xs:element name="dest" type="xs:anyURI"
            maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:attribute name="seviceUri" type="xs:anyURI"
          use="required"/>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

  <xs:element name="routingInformation" type="ri:riType"/>
  <xs:complexType name="riType">
    <xs:complexContent>
      <xs:restriction base="xs:anyType">
        <xs:sequence>
          <xs:element name="service" type="ri:service"
            maxOccurs="unbounded"/>
          <xs:any namespace="##other" processContents="lax"
            minOccurs="0" maxOccurs="unbounded"/>
        </xs:sequence>
        <xs:anyAttribute namespace="##any" processContents="lax"/>
      </xs:restriction>
    </xs:complexContent>
  </xs:complexType>

</xs:schema>
```

## 6. Examples

Figure 3 illustrates a <locationRequest> example that contains IP flow information in the request.

```
<locationRequest xmlns="urn:ietf:params:xml:ns:geopriv:held"
  responseType="emergencyRouting">

  <requestRoutingInformation
    xmlns="urn:ietf:params:xml:ns:geopriv:held:ri"/>

  <flow xmlns="urn:ietf:params:xml:ns:geopriv:held:flow"
    layer4="tcp" layer3="ipv4">
    <src>
      <address>192.168.1.1</address>
      <port>1024</port>
    </src>
    <dst>
      <address>10.0.0.1</address>
      <port>80</port>
    </dst>
  </flow>
</locationRequest>
```

Figure 3: Example Location Request.

Figure 4 illustrates the <locationResponse> message containing two location URIs: a HTTPS and a SIP URI. Additionally, the response contains routing information.

```
<locationResponse xmlns="urn:ietf:params:xml:ns:geopriv:held">
  <locationUriSet expires="2006-01-01T13:00:00.0Z">
    <locationURI>
      https://ls.example.com:9768/357yc6s64ceyoiuy5ax3o
    </locationURI>
    <locationURI>
      sip:9769+357yc6s64ceyoiuy5ax3o@ls.example.com
    </locationURI>
  </locationUriSet>

  <routingInformation
    xmlns="urn:ietf:params:xml:ns:geopriv:held:ri">
    <service serviceUri="urn:service:sos:police">
      <dest>sip:nypd@example.com</dest>
      <dest>sips:nypd@example.com</dest>
      <dest>xmpp:nypd@example.com</dest>
    </service>

    <service serviceUri="urn:service:sos:fire">
      <dest>sip:fd@ny.example.com</dest>
      <dest>sips:fd@ny.example.com</dest>
      <dest>xmpp:fd@ny.example.com</dest>
    </service>
  </routingInformation>

</locationResponse>
```

Figure 4: Example Location Response

## 7. Privacy Considerations

This document makes no changes that require privacy considerations beyond those already described in [RFC5985] and [RFC6155].

## 8. Security Considerations

This document imposes no additional security considerations beyond those already described in [RFC5985] and [RFC6155].

## 9. IANA Considerations

### 9.1. URN sub-namespace registration for 'urn:ietf:params:xml:ns:geopriv:held:ri'

This document calls for IANA to register a new XML namespace, as per the guidelines in [RFC3688].

URI: urn:ietf:params:xml:ns:geopriv:held:ri

Registrant Contact: IETF, ECRIT working group (ecrit@ietf.org),  
James Winterbottom (a.james.winterbottom@gmail.com).

XML:

BEGIN

```
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>HELD Routing Information Extensions</title>
  </head>
  <body>
    <h1>Additional Element for HELD Routing Information</h1>
    <h2>urn:ietf:params:xml:ns:geopriv:held:ri</h2>
    [[NOTE TO IANA/RFC-EDITOR: Please update RFC URL and replace XXXX
      with the RFC number for this specification.]]
    <p>See <a href="[[RFC URL]]">RFCXXXX</a>.</p>
  </body>
</html>
```

END

### 9.2. XML Schema Registration

This section registers an XML schema as per the procedures in [RFC3688].

URI: urn:ietf:params:xml:ns:geopriv:held:ri

Registrant Contact: IETF, ECRIT working group, (ecrit@ietf.org),  
James Winterbottom (a.james.winterbottom@gmail.com).

The XML for this schema can be found as the entirety of Section 5 of this document.

## 10. Acknowledgements

We would like to thank Wilfried Lange for sharing his views with us. We would also like to thank Bruno Chatras for his early review comments and Bernd Henschel for his support.

## 11. References

### 11.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.
- [RFC5222] Hardie, T., Newton, A., Schulzrinne, H., and H. Tschofenig, "LoST: A Location-to-Service Translation Protocol", RFC 5222, August 2008.
- [RFC5687] Tschofenig, H. and H. Schulzrinne, "GEOPRIV Layer 7 Location Configuration Protocol: Problem Statement and Requirements", RFC 5687, March 2010.
- [RFC5985] Barnes, M., "HTTP-Enabled Location Delivery (HELD)", RFC 5985, September 2010.
- [RFC6443] Rosen, B., Schulzrinne, H., Polk, J., and A. Newton, "Framework for Emergency Calling Using Internet Multimedia", RFC 6443, December 2011.
- [RFC6881] Rosen, B. and J. Polk, "Best Current Practice for Communications Services in Support of Emergency Calling", BCP 181, RFC 6881, March 2013.

### 11.2. Informative References

- [RFC5986] Thomson, M. and J. Winterbottom, "Discovering the Local Location Information Server (LIS)", RFC 5986, September 2010.
- [RFC6155] Winterbottom, J., Thomson, M., Tschofenig, H., and R. Barnes, "Use of Device Identity in HTTP-Enabled Location Delivery (HELD)", RFC 6155, March 2011.
- [RFC6915] Bellis, R., "Flow Identity Extension for HTTP-Enabled Location Delivery (HELD)", RFC 6915, April 2013.

[RFC7216] Thomson, M. and R. Bellis, "Location Information Server (LIS) Discovery Using IP Addresses and Reverse DNS", RFC 7216, April 2014.

Authors' Addresses

James Winterbottom  
Winterb Consulting Services  
Gwynneville, NSW 2500  
AU

Phone: +61 448 266004  
Email: a.james.winterbottom@gmail.com

Hannes Tschofenig  
Halls in Tirol 6060  
Austria

Phone:  
Email: Hannes.Tschofenig@gmx.net  
URI: <http://www.tschofenig.priv.at>

Laura Liess  
Deutsche Telekom Networks  
Deutsche Telekom Allee 7  
Darmstadt, Hessen 64295  
Germany

Phone:  
Email: L.Liess@telekom.de  
URI: <http://www.telekom.de>

