

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.

4.1.2. 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.

4.1.3. 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	http://www.nena.org
EENA	European Emergency Number Association	http://www.eena.org
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.

4.1.9. 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.

4.1.10. 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>
    </dm:device>
    <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>.
- [nc911] North Carolina 911 Board, "North Carolina Telecommunicator Reference", January 2009, <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>
<!-- 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

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

Phone: +1 724 382 1051
Email: 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
URI: <http://www.telecomsys.com>

James Winterbottom
AU

Email: 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.

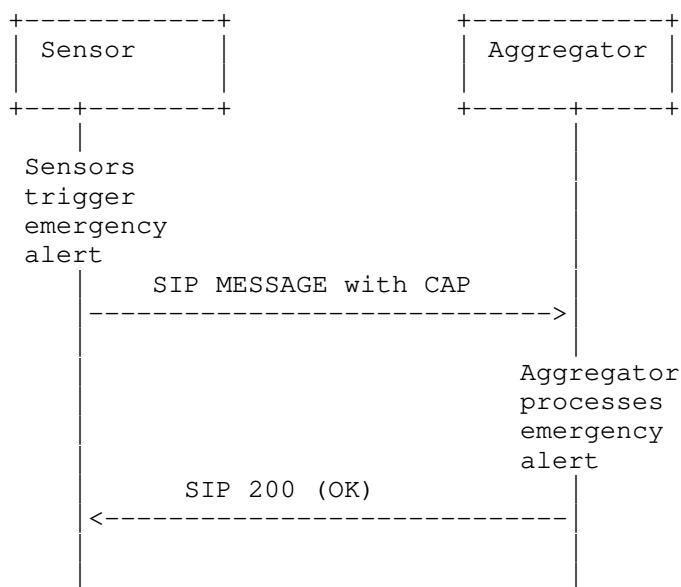


Figure 1: Targeted Emergency Alert Routing

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.

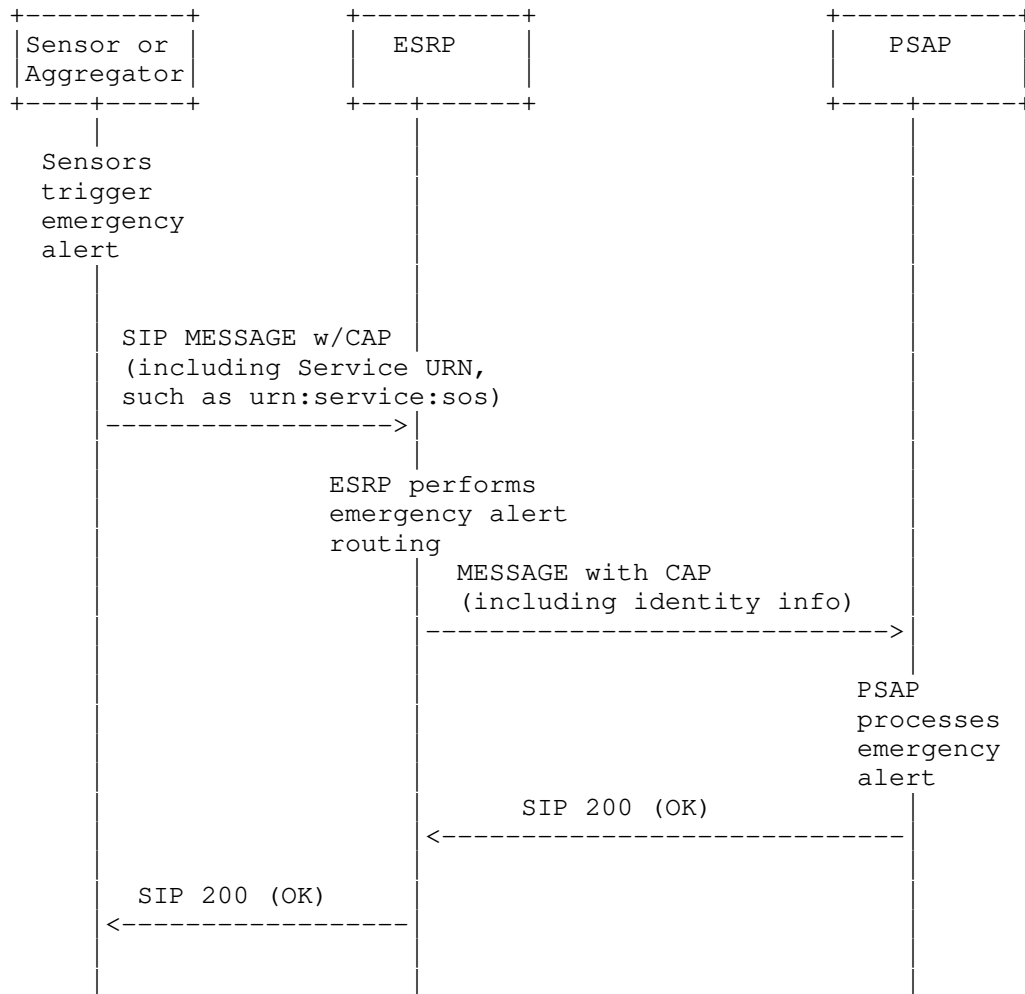


Figure 2: Location-Based Emergency Alert Routing

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>
  </info>
</alert>
```

```

    <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

Person and email address to contact for further information: Hannes Tschofenig, 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

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
URI: <http://www.cs.columbia.edu>

Hannes Tschofenig
ARM Limited

Austria

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

Randall Gellens
Core Technology Consulting

Email: rg+ietf@coretechnologyconsulting.com
URI: <http://www.coretechnologyconsulting.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.

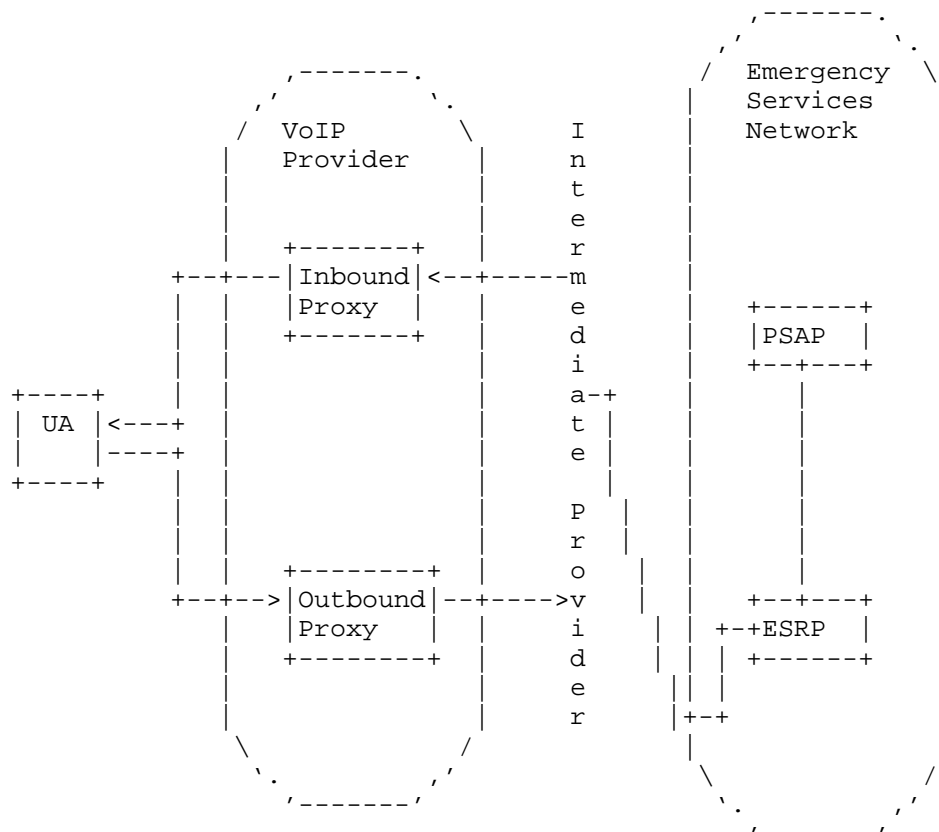


Figure 1: Example for Routing Asymmetry.

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.

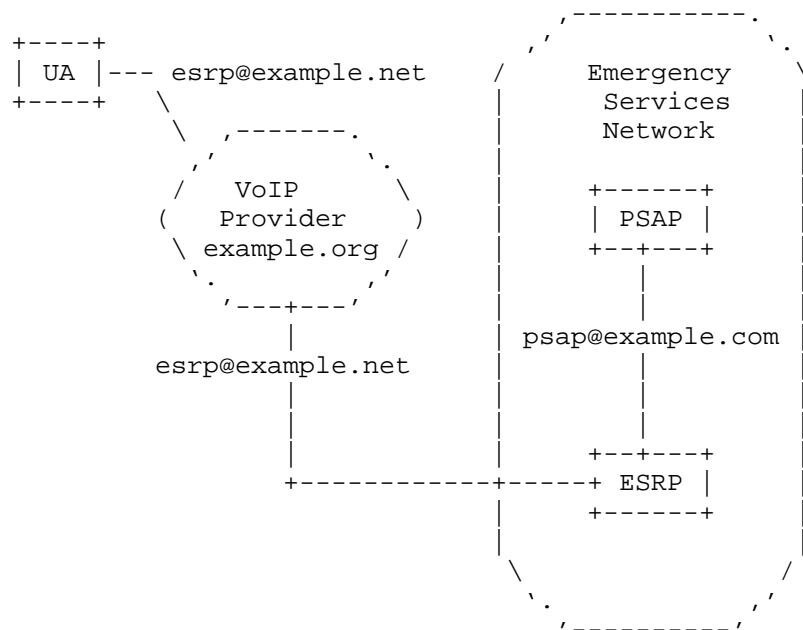


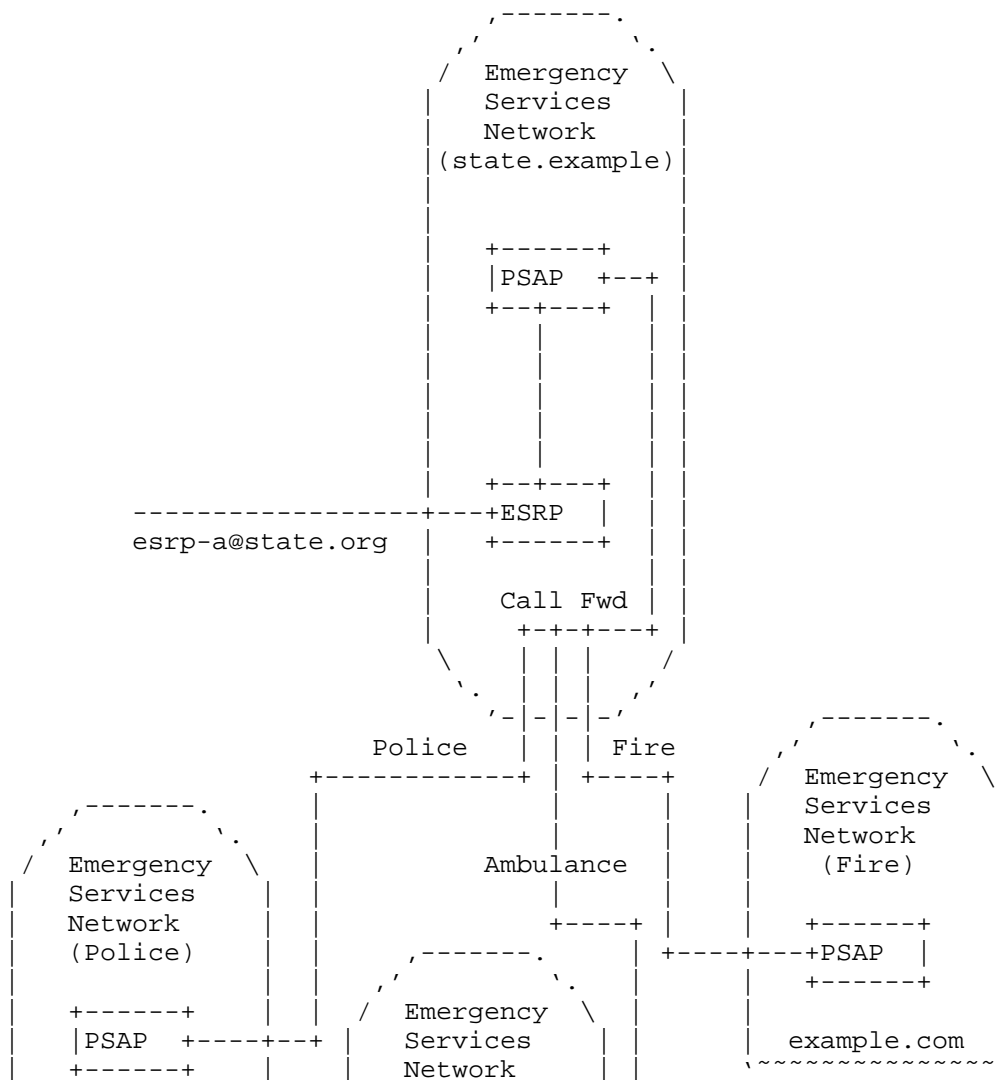
Figure 2: Example for Multi-Stage Routing.

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.



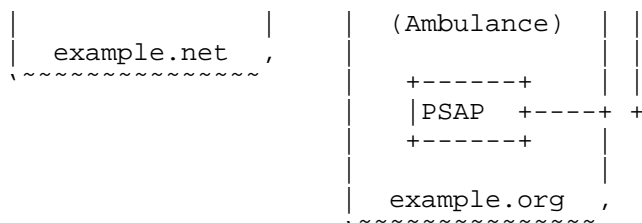


Figure 3: Example for Call Forwarding.

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.

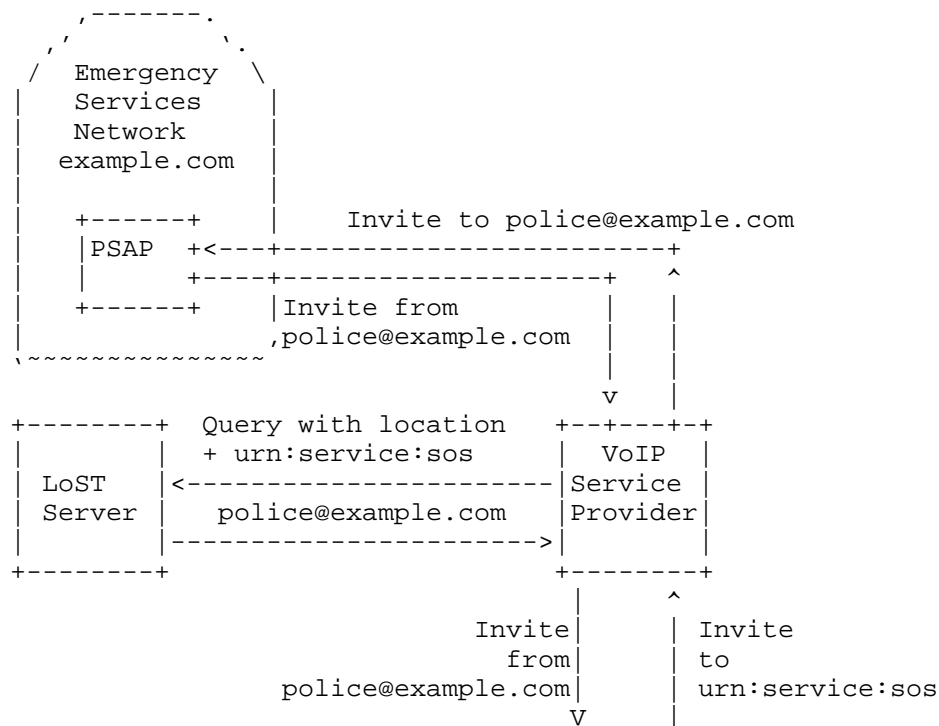




Figure 4: Example for Network-based Service URN Resolution.

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.

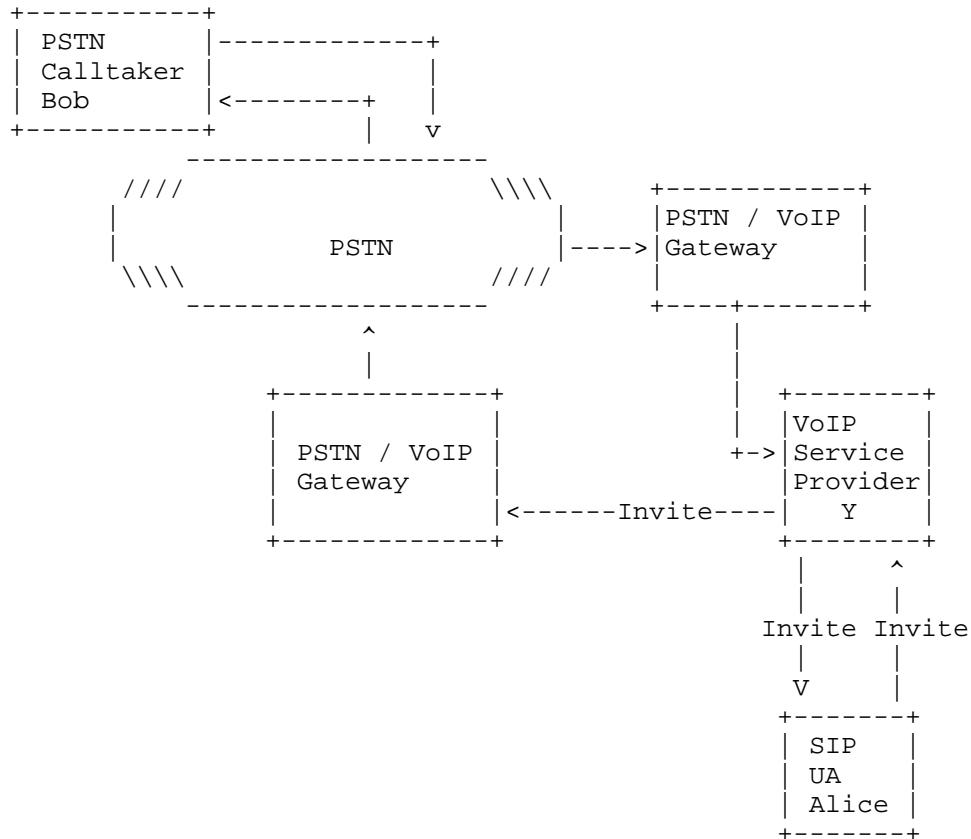


Figure 5: Example for PSTN Interworking.

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.

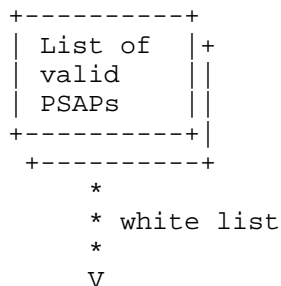
Furthermore, if the newly defined extension is not recognized, not verified adequately, or not obeyed by SIP intermediaries or SIP endpoints then it must not lead to a failure of the call handling procedure. Such call must be treated like a call that does not have any marking attached.

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.



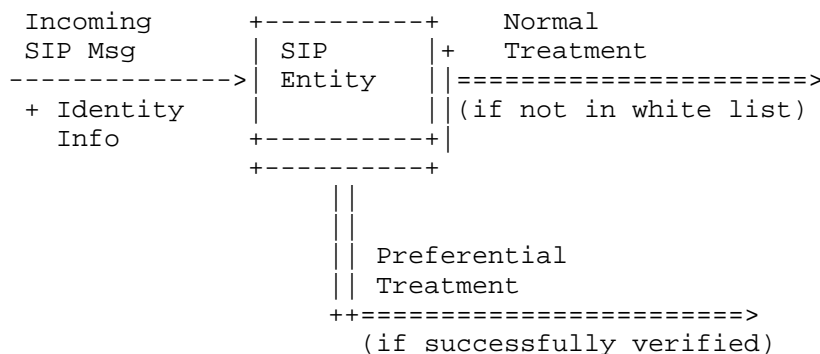


Figure 7: Identity-based Authorization

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
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
URI: <http://www.tschofenig.priv.at>

Christer Holmberg
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

EMail: christer.holmberg@ericsson.com

Milan Patel
InterDigital Communications

EMail: 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 Working Group
INTERNET-DRAFT
Category: Informational
Expires: January 5, 2015

H. Tschofenig
Independent
H. Schulzrinne
Columbia University
B. Aboba (ed.)
Microsoft Corporation
28 July 2014

Trustworthy Location
draft-ietf-ecrit-trustworthy-location-14.txt

Abstract

The trustworthiness of location information is critically important for some location-based applications, such as emergency calling or roadside assistance.

This document describes threats relating to conveyance of location in an emergency call, and describes techniques that improve the reliability and security of location information conveyed in a IP-based emergency service call. It also provides guidelines for assessing the trustworthiness of location information.

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 5, 2015.

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
1.1. Terminology	3
1.2. Emergency Services Architecture	5
2. Threat Models	8
2.1. Existing Work	8
2.2. Adversary Model	9
2.3. Location Spoofing	10
2.4. Identity Spoofing	10
3. Mitigation Techniques	11
3.1. Signed Location-by-Value	11
3.2. Location-by-Reference	15
3.3. Proxy Adding Location	18
4. Location Trust Assessment	19
5. Security Considerations	22
6. Privacy Considerations	23
7. IANA Considerations	25
8. References	25
8.1. Informative references	25
Acknowledgments	28
Authors' Addresses	29

1. Introduction

Several public and commercial services depend upon location information in their operations. This includes emergency services (such as fire, ambulance and police) as well as commercial services such as food delivery and roadside assistance.

For circuit-switched calls from landlines, as well as for Voice over IP (VoIP) services only supporting emergency service calls from stationary devices, location provided to the Public Safety Answering Point (PSAP) is determined from a lookup using the calling telephone number. As a result, for landlines or stationary VoIP, spoofing of caller identification can result in the PSAP incorrectly determining the caller's location. Problems relating to calling party number and Caller ID assurance have been analyzed by the "Secure Telephone Identity Revisited" [STIR] Working Group as described in "Secure Telephone Identity Problem Statement and Requirements" [I-D.ietf-stir-problem-statement]. In addition to the work underway in STIR, other mechanisms exist for validating caller identification. For example, as noted in [EENA], one mechanism for validating caller identification information (as well as the existence of an emergency) is for the PSAP to call the user back, as described in [RFC7090].

Given the existing work on caller identification, this document focuses on the additional threats that are introduced by the support of IP-based emergency services in nomadic and mobile devices, in which location may be conveyed to the PSAP within the emergency call. Ideally, a call taker at a PSAP should be able to assess, in real-time, the level of trust that can be placed on the information provided within a call. This includes automated location conveyed along with the call and location information communicated by the caller, as well as identity information relating to the caller or the device initiating the call. Where real-time assessment is not possible, it is important to be able to determine the source of the call in a post-incident investigation, so as to be able to enforce accountability.

This document defines terminology (including the meaning of "trustworthy location") in Section 1.1, reviews existing work in Section 1.2, describes the threat model in Section 2, outlines potential mitigation techniques in Section 3, covers trust assessment in Section 4 and discusses security considerations in Section 5.

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].

The definitions of "Internet Access Provider (IAP)", "Internet Service Provider (ISP)" and "Voice Service Provider (VSP)" are taken from "Requirements for Emergency Context Resolution with Internet Technologies" [RFC5012].

The definition of a "hoax call" is taken from "False Emergency Calls" [EENA].

The definition of "Device", "Target" and "Location Information Server" (LIS) is taken from "An Architecture for Location and Location Privacy in Internet Applications" [RFC6280], Section 7.

The term "Device" denotes the physical device, such as a mobile phone, PC, or embedded micro-controller, whose location is tracked as a proxy for the location of a Target.

The term "Target" denotes an individual or other entity whose location is sought in the Geopriv architecture. In many cases, the Target will be the human user of a Device, or it may be an object such as a vehicle or shipping container to which a Device is attached. In some instances, the Target will be the Device itself. The Target is the entity whose privacy Geopriv seeks to protect.

The term "Location Information Server" denotes an entity responsible for providing devices within an access network with information about their own locations. A Location Information Server uses knowledge of the access network and its physical topology to generate and distribute location information to devices.

The term "location determination method" refers to the mechanism used to determine the location of a Target. This may be something employed by a location information server (LIS), or by the Target itself. It specifically does not refer to the location configuration protocol (LCP) used to deliver location information either to the Target or the Recipient. This term is re-used from "GEOPRIV PIDF-LO Usage Clarification, Considerations, and Recommendations" [RFC5491].

The term "source" is used to refer to the LIS, node, or device from which a Recipient (Target or Third-Party) obtains location information.

Additionally, the terms Location-by-Value (LbyV), Location-by-Reference (LbyR), Location Configuration Protocol, Location Dereference Protocol, and Location Uniform Resource Identifier (URI) are re-used from "Requirements for a Location-by-Reference Mechanism" [RFC5808].

"Trustworthy Location" is defined as location information that can be

attributed to a trusted source, has been protected against modification in transmit, and has been assessed as trustworthy.

"Location Trust Assessment" refers to the process by which the reliability of location information can be assessed. This topic is discussed in Section 4.

"Identity Spoofing" is where the attacker forges or obscures their identity so as to prevent themselves from being identified as the source of the attack. One class of identity spoofing attack involves the forging of call origin identification.

The following additional terms apply to location spoofing:

"Place Shifting" is where the attacker constructs a Presence Information Data Format Location Object (PIDF-LO) for a location other than where they are currently located. In some cases, place shifting can be limited in range (e.g., within the coverage area of a particular cell tower).

"Time Shifting" is where the attacker uses or re-uses location information that was valid in the past, but is no longer valid because the attacker has moved.

"Location Theft" is where the attacker captures a Target's location information (possibly including a signature) and presents it as their own. Location theft can occur in a single instance, or may be continuous (e.g., where the attacker has gained control over the victim's device). Location theft may also be combined with time shifting to present someone else's location information after the original Target has moved.

1.2. Emergency Services Architecture

This section describes how location is utilized in the Internet Emergency Services Architecture, as well as the existing work on the problem of hoax calls.

1.2.1. Location

The Internet architecture for emergency calling is described in "Framework for Emergency Calling Using Internet Multimedia" [RFC6443]. Best practices for utilizing the architecture to make emergency calls are described in "Best Current Practice for Communications Services in Support of Emergency Calling" [RFC6881].

As noted in "An Architecture for Location and Location Privacy in Internet Applications" [RFC6280] Section 6.3:

"there are three critical steps in the placement of an emergency call, each involving location information:

1. Determine the location of the caller.
2. Determine the proper Public Safety Answering Point (PSAP) for the caller's location.
3. Send a SIP INVITE message, including the caller's location, to the PSAP."

The conveyance of location information within the Session Initiation Protocol (SIP) is described in "Location Conveyance for the Session Initiation Protocol" [RFC6442]. Conveyance of Location-by-Value (LbyV) as well as Location-by-Reference (LbyR) are supported. The Security Considerations (Section 7) discusses privacy, authentication and integrity concerns relating to conveyed location. This includes discussion of transmission layer security for confidentiality and integrity protection of SIP, as well as (undeployed) end-to-end security mechanisms for protection of location information (e.g. S/MIME). Regardless of whether transmission-layer security is utilized, location information may be available for inspection by an intermediary which, if it decides that the location value is unacceptable or insufficiently accurate, may send an error indication or replace the location, as described in [RFC6442] Section 3.4.

Although the infrastructure for location-based routing described in [RFC6443] was developed for use in emergency services, [RFC6442] supports conveyance of location within non-emergency calls as well as emergency calls. "Implications of 'retransmission-allowed' for SIP Location Conveyance" [RFC5606] Section 1 describes the overall architecture, as well as non-emergency usage scenarios:

The Presence Information Data Format for Location Objects (PIDF-LO [RFC4119]) carries both location information (LI) and policy information set by the Rule Maker, as is stipulated in [RFC3693]. The policy carried along with LI allows the Rule Maker to restrict, among other things, the duration for which LI will be retained by recipients and the redistribution of LI by recipients.

The Session Initiation Protocol [RFC3261] is one proposed Using Protocol for PIDF-LO. The conveyance of PIDF-LO within SIP is specified in [RFC6442]. The common motivation for providing LI in SIP is to allow location to be considered in routing the SIP message. One example use case would be emergency services, in which the location will be used by dispatchers to direct the response. Another use case might be providing location to be used by services associated with the SIP session; a location associated

with a call to a taxi service, for example, might be used to route to a local franchisee of a national service and also to route the taxi to pick up the caller.

1.2.2. Hoax Calls

Hoax calls have been a problem for emergency services dating back to the time of street corner call boxes. As the European Emergency Number Association (EENA) has noted [EENA]: "False emergency calls divert emergency services away from people who may be in life-threatening situations and who need urgent help. This can mean the difference between life and death for someone in trouble."

EENA [EENA] has attempted to define terminology and describe best current practices for dealing with false emergency calls. Reducing the number of hoax calls represents a challenge, since emergency services authorities in most countries are required to answer every call (whenever possible). Where the caller cannot be identified, the ability to prosecute is limited.

A particularly dangerous form of hoax call is "swatting" - a hoax emergency call that draws a response from law enforcement prepared for a violent confrontation (e.g. a fake hostage situation that results in dispatching of a "Special Weapons And Tactics" (SWAT) team). In 2008 the Federal Bureau of Investigation (FBI) issued a warning [Swatting] about an increase in the frequency and sophistication of these attacks.

As noted in [EENA], many documented cases of "swatting" involve not only the faking of an emergency, but also falsification or obfuscation of identity. There are a number of techniques by which hoax callers attempt to avoid identification, and in general, the ability to identify the caller appears to influence the incidence of hoax calls.

Where a Voice Service Provider enables setting of the outbound caller identification without checking it against the authenticated identity, forging caller identification is trivial. Similarly where an attacker can gain entry to a Private Branch Exchange (PBX), they can then subsequently use that access to launch a denial of service attack against the PSAP, or to make fraudulent emergency calls. Where emergency calls have been allowed from handsets lacking a SIM card, or where ownership of the SIM card cannot be determined, the frequency of hoax calls has often been unacceptably high [TASMANIA][UK][SA].

However, there are few documented cases of hoax calls that have arisen from conveyance of untrustworthy location information within

an emergency call, which is the focus of this document.

2. Threat Models

This section reviews existing analyses of the security of emergency services, threats to geographic location privacy, threats relating to spoofing of caller identification and modification of location information in transit. In addition, the threat model applying to this work is described.

2.1. Existing Work

"An Architecture for Location and Location Privacy in Internet Applications" [RFC6280] describes an architecture for privacy-preserving location-based services in the Internet, focusing on authorization, security and privacy requirements for the data formats and protocols used by these services.

Within the Security Considerations (Section 5), mechanisms for ensuring the security of the location distribution chain are discussed; these include mechanisms for hop-by-hop confidentiality and integrity protection as well as end-to-end assurance.

"Geopriv Requirements" [RFC3693] focuses on the authorization, security and privacy requirements of location-dependent services, including emergency services. Within the Security Considerations (Section 8), this includes discussion of emergency services authentication (Section 8.3), and issues relating to identity and anonymity (Section 8.4).

"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.

"Security Threats and Requirements for Emergency Call Marking and Mapping" [RFC5069] reviews security threats associated with the marking of signaling messages and the process of mapping locations to Universal Resource Identifiers (URIs) that point to PSAPs. RFC 5069 describes 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. In addition, it describes attacks against individuals, including attempts to prevent an individual from receiving aid, or to gain information about an emergency, as well as attacks on emergency services infrastructure elements, such as mapping discovery and mapping servers.

"Secure Telephone Identity Threat Model" [I-D.ietf-stir-threats] analyzes threats relating to impersonation and obscuring of calling party numbers, reviewing the capabilities available to attackers, and the scenarios in which attacks are launched.

2.2. Adversary Model

To provide a structured analysis we distinguish between three adversary models:

External adversary model: The end host, e.g., an emergency caller whose location is going to be communicated, is honest and the adversary may be located between the end host and the location server or between the end host and the PSAP. None of the emergency service infrastructure elements act maliciously.

Malicious infrastructure adversary model: The emergency call routing elements, such as the Location Information Server (LIS), the Location-to-Service Translation (LoST) infrastructure, used for mapping locations to PSAP address, or call routing elements, may act maliciously.

Malicious end host adversary model: The end host itself acts maliciously, whether the owner is aware of this or whether it is acting under the control of a third party.

Since previous work describes attacks against infrastructure elements (e.g. location servers, call route servers, mapping servers) or the emergency services IP network, as well as threats from attackers attempting to snoop location in transit, this document focuses on the threats arising from end hosts providing false location information within emergency calls (the malicious end host adversary model).

Since the focus is on malicious hosts, we do not cover threats that may arise from attacks on infrastructure that hosts depend on to obtain location. For example, end hosts may obtain location from civilian GPS, which is vulnerable to spoofing [GPSCounter] or from third party Location Service Providers (LSPs) which may be vulnerable to attack or may not provide location accuracy suitable for emergency purposes.

Also, we do not cover threats arising from inadequate location infrastructure. For example, a stale wiremap or an inaccurate access point location database could be utilized by the Location Information Server (LIS) or the end host in its location determination, thereby leading to an inaccurate determination of location. Similarly, a Voice Service Provider (VSP) (and indirectly a LIS) could utilize the wrong identity (such as an IP address) for location lookup, thereby

providing the end host with misleading location information.

2.3. Location Spoofing

Where location is attached to the emergency call by an end host, the end host can fabricate a PIDF-LO and convey it within an emergency call. The following represent examples of location spoofing:

Place shifting: Trudy, the adversary, pretends to be at an arbitrary location.

Time shifting: Trudy pretends to be at a location she was a while ago.

Location theft: Trudy observes or obtains Alice's location and replays it as her own.

2.4. Identity Spoofing

While this document does not focus on the problems created by determination of location based on spoofed caller identification, the ability to ascertain identity is important, since the threat of punishment reduces hoax calls. As an example, calls from pay phones are subject to greater scrutiny by the call taker.

With calls originating on an IP network, at least two forms of identity are relevant, with the distinction created by the split between the IAP and the VSP:

(a) network access identity such as might be determined via authentication (e.g., using the Extensible Authentication Protocol (EAP) [RFC3748]);

(b) caller identity, such as might be determined from authentication of the emergency caller at the VoIP application layer.

If the adversary did not authenticate itself to the VSP, then accountability may depend on verification of the network access identity. However, this also may not have been authenticated, such as in the case where an open IEEE 802.11 Access Point is used to initiate a hoax emergency call. Although endpoint information such as the IP or MAC address may have been logged, tying this back to the device owner may be challenging.

Unlike the existing telephone system, VoIP emergency calls can provide an identity that need not necessarily be coupled to a business relationship with the IAP, ISP or VSP. However, due to the time-critical nature of emergency calls, multi-layer authentication

is undesirable, so that in most cases, only the device placing the call will be able to be identified. Furthermore, deploying additional credentials for emergency service purposes (such as certificates) increases costs, introduces a significant administrative overhead and is only useful if widely deployed.

3. Mitigation Techniques

The sections that follow present three mechanisms for mitigating the threats presented in Section 2:

1. Signed location by value (Section 3.1), which provides for authentication and integrity protection of the PIDF-LO. At the time of this writing, there is only an expired straw-man proposal for this mechanism [I-D.thomson-geopriv-location-dependability], so that it is not suitable for deployment.

2. Location-by-reference (Section 3.2), which enables location to be obtained by the PSAP directly from the location server, over a confidential and integrity-protected channel, avoiding modification by the end-host or an intermediary. This mechanism is specified in [RFC6753].

3. Proxy added location (Section 3.3), which protects against location forgery by the end host. This mechanism is specified in [RFC6442].

3.1. Signed Location-by-Value

With location signing, a location server signs the location information before it is sent to the Target. The signed location information is then sent to the location recipient, who verifies it.

Figure 1 shows the communication model with the target requesting signed location in step (a), the location server returns it in step (b) and it is then conveyed to the location recipient in step (c) who verifies it. For SIP, the procedures described in "Location Conveyance for the Session Initiation Protocol" [RFC6442] are applicable for location conveyance.

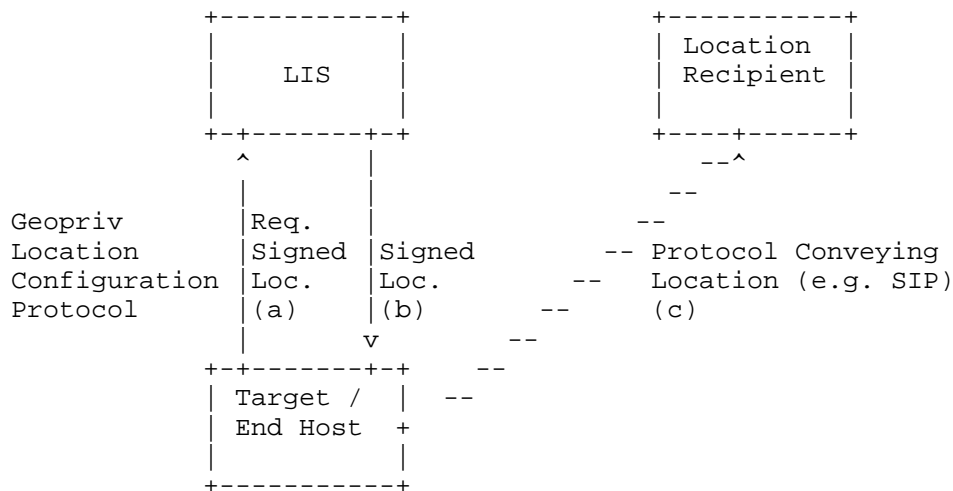


Figure 1: Location Signing

A straw-man proposal for location signing is provided in "Digital Signature Methods for Location Dependability" [I-D.thomson-geopriv-location-dependability]. Note that since this document is no longer under development, location signing cannot be considered deployable at the time of this writing.

In order to limit replay attacks, this document proposes the addition of a "validity" element to the PIDF-LO, including a "from" sub-element containing the time that location information was validated by the signer, as well as an "until" sub-element containing the last time that the signature can be considered valid.

One of the consequences of including an "until" element is that even a stationary target would need to periodically obtain a fresh PIDF-LO, or incur the additional delay of querying during an emergency call.

Although privacy-preserving procedures may be disabled for emergency calls, by design, PIDF-LO objects limit the information available for real-time attribution. As noted in [RFC5985] Section 6.6:

The LIS MUST NOT include any means of identifying the Device in the PIDF-LO unless it is able to verify that the identifier is correct and inclusion of identity is expressly permitted by a Rule Maker. Therefore, PIDF parameters that contain identity are either omitted or contain unlinked pseudonyms [RFC3693]. A unique, unlinked presentity URI SHOULD be generated by the LIS for the mandatory presence "entity" attribute of the PIDF document.

Optional parameters such as the "contact" and "deviceID" elements [RFC4479] are not used.

Also, the device referred to in the PIDF-LO may not necessarily be the same entity conveying the PIDF-LO to the PSAP. As noted in [RFC6442] Section 1:

In no way does this document assume that the SIP user agent client that sends a request containing a location object is necessarily the Target. The location of a Target conveyed within SIP typically corresponds to that of a device controlled by the Target, for example, a mobile phone, but such devices can be separated from their owners, and moreover, in some cases, the user agent may not know its own location.

Without the ability to tie the target identity to the identity asserted in the SIP message, it is possible for an attacker to cut and paste a PIDF-LO obtained by a different device or user into a SIP INVITE and send this to the PSAP. This cut and paste attack could succeed even when a PIDF-LO is signed, or [RFC4474] is implemented.

To address location-spoofing attacks, [I-D.thomson-geopriv-location-dependability] proposes addition of an "identity" element which could include a SIP URI (enabling comparison against the identity asserted in the SIP headers) or an X.509v3 certificate. If the target was authenticated by the LIS, an "authenticated" attribute is added. However, inclusion of an "identity" attribute could enable location tracking, so that a "hash" element is also proposed which could contain a hash of the content of the "identity" element instead. In practice, such a hash would not be much better for real-time validation than a pseudonym.

Location signing cannot deter attacks in which valid location information is provided. For example, an attacker in control of compromised hosts could launch a denial-of-service attack on the PSAP by initiating a large number of emergency calls, each containing valid signed location information. Since the work required to verify the location signature is considerable, this could overwhelm the PSAP infrastructure.

However, while DDOS attacks are unlikely to be deterred by location signing, accurate location information would limit the subset of compromised hosts that could be used for an attack, as only hosts within the PSAP serving area would be useful in placing emergency calls.

Location signing is also difficult when the host obtains location via mechanisms such as GPS, unless trusted computing approaches, with

tamper-proof GPS modules, can be applied. Otherwise, an end host can pretend to have a GPS device, and the recipient will need to rely on its ability to assess the level of trust that should be placed in the end host location claim.

Even though location signing mechanisms have not been standardized, [NENA-i2] Section 3.7 includes operational recommendations relating to location signing:

Location determination is out of scope for NENA, but we can offer guidance on what should be considered when designing mechanisms to report location:

1. The location object should be digitally signed.
2. The certificate for the signer (LIS operator) should be rooted in VESA. For this purpose, VPC and ERDB operators should issue certs to LIS operators.
3. The signature should include a timestamp.
4. Where possible, the Location Object should be refreshed periodically, with the signature (and thus the timestamp) being refreshed as a consequence.
5. Anti-spoofing mechanisms should be applied to the Location Reporting method.

[Note: The term Valid Emergency Services Authority (VESA) refers to the root certificate authority. VPC stands for VoIP Positioning Center and ERDB stands for the Emergency Service Zone Routing Database.]

As noted above, signing of location objects implies the development of a trust hierarchy that would enable a certificate chain provided by the LIS operator to be verified by the PSAP. Rooting the trust hierarchy in VESA can be accomplished either by having the VESA directly sign the LIS certificates, or by the creation of intermediate Certificate Authorities (CAs) certified by the VESA, which will then issue certificates to the LIS. In terms of the workload imposed on the VESA, the latter approach is highly preferable. However, this raises the question of who would operate the intermediate CAs and what the expectations would be.

In particular, the question arises as to the requirements for LIS certificate issuance, and how they would compare to requirements for issuance of other certificates such as an SSL/TLS web certificate.

3.2. Location-by-Reference

Location-by-Reference was developed so that end hosts can avoid having to periodically query the location server for up-to-date location information in a mobile environment. Additionally, if operators do not want to disclose location information to the end host without charging them, location-by-reference provides a reasonable alternative. Also, since location-by-reference enables the PSAP to directly contact the location server, it avoids potential attacks by intermediaries.

As noted in "A Location Dereference Protocol Using HTTP-Enabled Location Delivery (HELD)" [RFC6753], a location reference can be obtained via HTTP-Enabled Location Delivery (HELD) [RFC5985]. In addition, "Location Configuration Extensions for Policy Management" [RFC7199] extends location configuration protocols such as HELD to provide hosts with a reference to the rules that apply to a Location-by-Reference so that the host can view or set these rules.

Figure 2 shows the communication model with the target requesting a location reference in step (a), the location server returns the reference and potentially the policy in step (b), and it is then conveyed to the location recipient in step (c). The location recipient needs to resolve the reference with a request in step (d). Finally, location information is returned to the Location Recipient afterwards. For location conveyance in SIP, the procedures described in [RFC6442] are applicable.

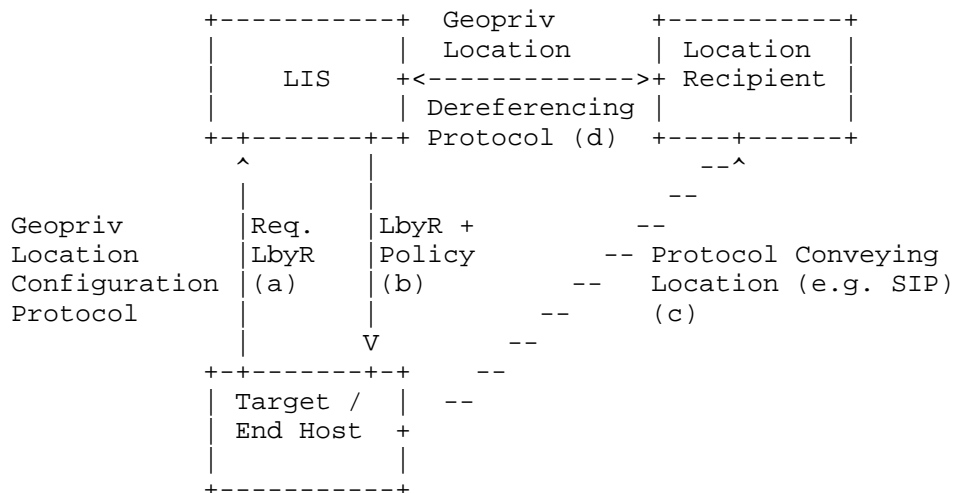


Figure 2: Location by Reference

Where location by reference is provided, the recipient needs to deference the LbyR in order to obtain location. The details for the dereferencing operations vary with the type of reference, such as a HTTP, HTTPS, SIP, SIPS URI or a SIP presence URI.

For location-by-reference, the location server needs to maintain one or several URIs for each target, timing out these URIs after a certain amount of time. References need to expire to prevent the recipient of such a Uniform Resource Locator (URL) from being able to permanently track a host and to offer garbage collection functionality for the location server.

Off-path adversaries must be prevented from obtaining the target's location. The reference contains a randomized component that prevents third parties from guessing it. When the location recipient fetches up-to-date location information from the location server, it can also be assured that the location information is fresh and not replayed. However, this does not address location theft.

With respect to the security of the de-reference operation, [RFC6753] Section 6 states:

TLS MUST be used for dereferencing location URIs unless confidentiality and integrity are provided by some other mechanism, as discussed in Section 3. Location Recipients MUST authenticate the host identity using the domain name included in the location URI, using the procedure described in Section 3.1 of [RFC2818]. Local policy determines what a Location Recipient does if authentication fails or cannot be attempted.

The authorization by possession model (Section 4.1) further relies on TLS when transmitting the location URI to protect the secrecy of the URI. Possession of such a URI implies the same privacy considerations as possession of the PIDF-LO document that the URI references.

Location URIs MUST only be disclosed to authorized Location Recipients. The GEOPRIV architecture [RFC6280] designates the Rule Maker to authorize disclosure of the URI.

Protection of the location URI is necessary, since the policy attached to such a location URI permits anyone who has the URI to view the associated location information. This aspect of security is covered in more detail in the specification of location conveyance protocols, such as [RFC6442].

For authorizing access to location-by-reference, two authorization models were developed: "Authorization by Possession" and

"Authorization via Access Control Lists". With respect to "Authorization by Possession" [RFC6753] Section 4.1 notes:

In this model, possession -- or knowledge -- of the location URI is used to control access to location information. A location URI might be constructed such that it is hard to guess (see C8 of [RFC5808]), and the set of entities that it is disclosed to can be limited. The only authentication this would require by the LS is evidence of possession of the URI. The LS could immediately authorize any request that indicates this URI.

Authorization by possession does not require direct interaction with Rule Maker; it is assumed that the Rule Maker is able to exert control over the distribution of the location URI. Therefore, the LIS can operate with limited policy input from a Rule Maker.

Limited disclosure is an important aspect of this authorization model. The location URI is a secret; therefore, ensuring that adversaries are not able to acquire this information is paramount. Encryption, such as might be offered by TLS [RFC5246] or S/MIME [RFC5751], protects the information from eavesdroppers.

Using possession as a basis for authorization means that, once granted, authorization cannot be easily revoked. Cancellation of a location URI ensures that legitimate users are also affected; application of additional policy is theoretically possible but could be technically infeasible. Expiration of location URIs limits the usable time for a location URI, requiring that an attacker continue to learn new location URIs to retain access to current location information.

In situations where "Authorization by Possession" is not suitable (such as where location hiding [RFC6444] is required), the "Authorization via Access Control Lists" model may be preferred.

Without the introduction of hierarchy, it would be necessary for the PSAP to obtain credentials, such as certificates or shared symmetric keys, for all the LISes in its coverage area, to enable it to successfully dereference LbyRs. In situations with more than a few LISes per PSAP, this would present operational challenges.

A certificate hierarchy providing PSAPs with client certificates chaining to the VESA could be used to enable the LIS to authenticate and authorize PSAPs for dereferencing. Note that unlike PIDF-LO signing (which mitigates against modification of PIDF-LOs), this merely provides the PSAP with access to a (potentially unsigned) PIDF-LO, albeit over a protected TLS channel.

Another approach would be for the local LIS to upload location information to a location aggregation point who would in turn manage the relationships with the PSAP. This would shift the management burden from the PSAPs to the location aggregation points.

3.3. Proxy Adding Location

Instead of relying upon the end host to provide location, is possible for a proxy that has the ability to determine the location of the end point (e.g., based on the end host IP or MAC address) to retrieve and add or override location information. This requires deployment of application layer entities by ISPs, unlike the two other techniques. The proxies could be used for emergency or non-emergency communications, or both.

The use of proxy-added location is primarily applicable in scenarios where the end host does not provide location. As noted in [RFC6442] Section 4.1:

A SIP intermediary SHOULD NOT add location to a SIP request that already contains location. This will quite often lead to confusion within LRs. However, if a SIP intermediary adds location, even if location was not previously present in a SIP request, that SIP intermediary is fully responsible for addressing the concerns of any 424 (Bad Location Information) SIP response it receives about this location addition and MUST NOT pass on (upstream) the 424 response. A SIP intermediary that adds a locationValue MUST position the new locationValue as the last locationValue within the Geolocation header field of the SIP request.

A SIP intermediary MAY add a Geolocation header field if one is not present -- for example, when a user agent does not support the Geolocation mechanism but their outbound proxy does and knows the Target's location, or any of a number of other use cases (see Section 3).

As noted in [RFC6442] Section 3.3:

This document takes a "you break it, you bought it" approach to dealing with second locations placed into a SIP request by an intermediary entity. That entity becomes completely responsible for all location within that SIP request (more on this in Section 4).

While it is possible for the proxy to override location included by the end host, [RFC6442] Section 3.4 notes the operational limitations:

Overriding location information provided by the user requires a deployment where an intermediary necessarily knows better than an end user -- after all, it could be that Alice has an on-board GPS, and the SIP intermediary only knows her nearest cell tower. Which is more accurate location information? Currently, there is no way to tell which entity is more accurate or which is wrong, for that matter. This document will not specify how to indicate which location is more accurate than another.

The disadvantage of this approach is the need to deploy application layer entities, such as SIP proxies, at IAPs or associated with IAPs. This requires a standardized VoIP profile to be deployed at every end device and at every IAP. This might impose interoperability challenges.

Additionally, the IAP needs to take responsibility for emergency calls, even for customers they have no direct or indirect relationship with. To provide identity information about the emergency caller from the VSP it would be necessary to let the IAP and the VSP to interact for authentication (see, for example, "Diameter Session Initiation Protocol (SIP) Application" [RFC4740]). This interaction along the Authentication, Authorization and Accounting infrastructure is often based on business relationships between the involved entities. An arbitrary IAP and VSP are unlikely to have a business relationship. In case the interaction between the IAP and the VSP fails due to the lack of a business relationship then typically a fall-back would be provided where no emergency caller identity information is made available to the PSAP and the emergency call still has to be completed.

4. Location Trust Assessment

The ability to assess the level of trustworthiness of conveyed location information is important, since this makes it possible to understand how much value should be placed on location information, as part of the decision making process. As an example, if automated location information is understood to be highly suspect or is absent, a call taker can put more effort into verifying the authenticity of the call and to obtaining location information from the caller.

Location trust assessment has value regardless of whether the location itself is authenticated (e.g. signed location) or is obtained directly from the location server (e.g. location-by-reference) over security transport, since these mechanisms do not provide assurance of the validity or provenance of location data.

To prevent location-theft attacks, the "entity" element of the PIDF-LO is of limited value if an unlinked pseudonym is provided in this

field. However, if the LIS authenticates the target, then the linkage between the pseudonym and the target identity can be recovered in a post-incident investigation.

As noted in [I.D.thomson-geopriv-location-dependability], if the location object was signed, the location recipient has additional information on which to base their trust assessment, such as the validity of the signature, the identity of the target, the identity of the LIS, whether the LIS authenticated the target, and the identifier included in the "entity" field.

Caller accountability is also an important aspect of trust assessment. Can the individual purchasing the device or activating service be identified or did the call originate from a non-service initialized (NSI) device whose owner cannot be determined? Prior to the call, was the caller authenticated at the network or application layer? In the event of a hoax call, can audit logs be made available to an investigator, or can information relating to the owner of an unlinked pseudonym be provided, enabling investigators to unravel the chain of events that lead to the attack?

In practice, the source of the location data is important for location trust assessment. For example, location provided by a Location Information Server (LIS) whose administrator has an established history of meeting emergency location accuracy requirements (e.g. Phase II) may be considered more reliable than location information provided by a third party Location Service Provider (LSP) that disclaims use of location information for emergency purposes.

However, even where an LSP does not attempt to meet the accuracy requirements for emergency location, it still may be able to provide information useful in assessing about how reliable location information is likely to be. For example, was location determined based on the nearest cell tower or 802.11 Access Point (AP), or was a triangulation method used? If based on cell tower or AP location data, was the information obtained from an authoritative source (e.g. the tower or AP owner) and when was the last time that the location of the tower or access point was verified?

For real-time validation, information in the signaling and media packets can be cross checked against location information. For example, it may be possible to determine the city, state, country or continent associated with the IP address included within SIP Via: or Contact: headers, or the media source address, and compare this against the location information reported by the caller or conveyed in the PIDF-LO. However, in some situations only entities close to the caller may be able to verify the correctness of location

information.

Real-time validation of the timestamp contained within PIDF-LO objects (reflecting the time at which the location was determined) is also challenging. To address time-shifting attacks, the "timestamp" element of the PIDF-LO, defined in [RFC3863], can be examined and compared against timestamps included within the enclosing SIP message, to determine whether the location data is sufficiently fresh. However, the timestamp only represents an assertion by the LIS, which may or may not be trustworthy. For example, the recipient of the signed PIDF-LO may not know whether the LIS supports time synchronization, or whether it is possible to reset the LIS clock manually without detection. Even if the timestamp was valid at the time location was determined, a time period may elapse between when the PIDF-LO was provided and when it is conveyed to the recipient. Periodically refreshing location information to renew the timestamp even though the location information itself is unchanged puts additional load on LISes. As a result, recipients need to validate the timestamp in order to determine whether it is credible.

While this document focuses on the discussion of real-time determination of suspicious emergency calls, the use of audit logs may help in enforcing accountability among emergency callers. For example, in the event of a hoax call, information relating to the owner of the unlinked pseudonym could be provided to investigators, enabling them to unravel the chain of events that lead to the attack. However, while auditability is an important deterrent, it is likely to be of most benefit in situations where attacks on the emergency services system are likely to be relatively infrequent, since the resources required to pursue an investigation are likely to be considerable. However, although real-time validation based on PIDF-LO elements is challenging, where LIS audit logs are available (such as where a law enforcement agency can present a subpoena), linking of a pseudonym to the device obtaining location can be accomplished during an investigation.

Where attacks are frequent and continuous, automated mechanisms are required. For example, it might be valuable to develop mechanisms to exchange audit trails information in a standardized format between ISPs and PSAPs / VSPs and PSAPs or heuristics to distinguish potentially fraudulent emergency calls from real emergencies. While a Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) may be applied to suspicious calls to lower the risk from bot-nets, this is quite controversial for emergency services, due to the risk of delaying or rejecting valid calls.

5. Security Considerations

Although it is important to ensure that location information cannot be faked, the mitigation techniques presented in this document are not universally applicable. For example, there will be many GPS-enabled devices that will find it difficult to utilize any of the solutions described in Section 3. It is also unlikely that users will be willing to upload their location information for "verification" to a nearby location server located in the access network.

This document focuses on threats that arise from conveyance of misleading location information, rather than caller identification or authentication and integrity protection of the messages in which location is conveyed. Nevertheless, these aspects are important. In some countries, regulators may not require the authenticated identity of the emergency caller (e.g. emergency calls placed from PSTN pay phones or SIM-less cell phones). Furthermore, if identities can easily be crafted (as it is the case with many VoIP offerings today), then the value of emergency caller authentication itself might be limited. As a result, attackers can forge emergency calls with a lower risk of being held accountable, which may encourage hoax calls.

In order to provide authentication and integrity protection for the Session Initiation Protocol (SIP) messages conveying location, several security approaches are available. It is possible to ensure that modification of the identity and location in transit can be detected by the location recipient (e.g., the PSAP), using cryptographic mechanisms, as described in "Enhancements for Authenticated Identity Management in the Session Initiation Protocol" [RFC4474]. However, compatibility with Session Border Controllers (SBCs) that modify integrity-protected headers has proven to be an issue in practice, and as a result, a revision is in progress [I.D.ietf-stir-rfc4474bis]. In the absence of an end-to-end solution, SIP over Transport Layer Security (TLS) can be used to provide message authentication and integrity protection hop-by-hop.

PSAPs remain vulnerable to distributed denial of service attacks, even where the mitigation techniques described in this document are utilized. Placing a large number of emergency calls that appear to come from different locations is an example of an attack that is difficult to carry out within the legacy system, but is easier to imagine within IP-based emergency services. Also, in the current system, it would be very difficult for an attacker from country 'Foo' to attack the emergency services infrastructure located in country 'Bar', but this attack is possible within IP-based emergency services.

While manually mounting the attacks described in Section 2 is non-trivial, the attacks described in this document can be automated. While manually carrying out a location theft would require the attacker to be in proximity to the location being spoofed, or to collude with another end host, an attacker able to run code on an end host can obtain its location, and cause an emergency call to be made. While manually carrying out a time shifting attack would require that the attacker visit the location and submit it before the location information is considered stale, while traveling rapidly away from that location to avoid apprehension, these limitations would not apply to an attacker able to run code on the end host. While obtaining a PIDF-LO from a spoofed IP address requires that the attacker be on the path between the HELD requester and the LIS, if the attacker is able to run code requesting the PIDF-LO, retrieve it from the LIS, and then make an emergency call using it, this attack becomes much easier. To mitigate the risk of automated attacks, service providers can limit the ability of untrusted code (such as WebRTC applications written in Javascript) to make emergency calls.

Emergency services have three finite resources subject to denial of service attacks: the network and server infrastructure, call takers and dispatchers, and the first responders, such as fire fighters and police officers. Protecting the network infrastructure is similar to protecting other high-value service providers, except that location information may be used to filter call setup requests, to weed out requests that are out of area. Even for large cities PSAPs may only have a handful of call takers on duty. So even if automated techniques are utilized to evaluate the trustworthiness of conveyed location and call takers can, by questioning the caller, eliminate many hoax calls, PSAPs can be overwhelmed even by a small-scale attack. Finally, first responder resources are scarce, particularly during mass-casualty events.

6. Privacy Considerations

The emergency calling architecture described in [RFC6443] utilizes the PIDF-LO format defined in [RFC4119]. As described in the location privacy architecture [RFC6280], privacy rules that may include policy instructions are conveyed along with the location object.

The intent of the location privacy architecture was to provide strong privacy protections, as noted in [RFC6280] Section 1.1:

A central feature of the Geopriv architecture is that location information is always bound to privacy rules to ensure that entities that receive location information are informed of how they may use it. These rules can convey simple directives ("do

not share my location with others"), or more robust preferences ("allow my spouse to know my exact location all of the time, but only allow my boss to know it during work hours")... The binding of privacy rules to location information can convey users' desire for and expectations of privacy, which in turn helps to bolster social and legal systems' protection of those expectations.

However, in practice this architecture has limitations which apply within emergency and non-emergency situations. As noted in Section 1.2.2, concerns about hoax calls have lead to restrictions on anonymous emergency calls. Caller identification (potentially asserted in SIP via P-Asserted-Identity and via SIP Identity) may be used during emergency calls. As a result, in many cases location information transmitted within SIP messages can be linked to caller identity. For example, in case of signed LbyV, there are privacy concerns arising from linking the location object to identifiers to prevent replay attacks, as described in Section 3.1.

The ability to observe location information during emergency calls may also represent a privacy risk. As a result, [RFC6443] requires transmission layer security for SIP messages, as well as interactions with the location server. However, even where transmission layer security is used, privacy rules associated with location information may not apply.

In many jurisdictions, an individual requesting emergency assistance is assumed to be granting permission to the PSAP, call taker and first responders to obtain their location in order to accelerate dispatch. As a result, privacy policies associated with location are implicitly waived when an emergency call is initiated. In addition, when location information is included within SIP messages either in emergency or non-emergency uses, SIP entities receiving the SIP message are implicitly assumed to be authorized location recipients, as noted in [RFC5606] Section 3.2:

Consensus has emerged that any SIP entity that receives a SIP message containing LI through the operation of SIP's normal routing procedures or as a result of location-based routing should be considered an authorized recipient of that LI. Because of this presumption, one SIP element may pass the LI to another even if the LO it contains has <retransmission-allowed> set to "no"; this sees the passing of the SIP message as part of the delivery to authorized recipients, rather than as retransmission. SIP entities are still enjoined from passing these messages outside the normal routing to external entities if <retransmission-allowed> is set to "no", as it is the passing to third parties that <retransmission-allowed> is meant to control.

Where LbyR is utilized rather than LbyV, it is possible to apply more restrictive authorization policies, limiting access to intermediaries and snoopers. However, this is not possible if the "authorization by possession" model is used.

7. IANA Considerations

This document does not require actions by IANA.

8. References

8.1. Informative References

[I-D.ietf-stir-problem-statement]

Peterson, J., Schulzrinne, H., and H. Tschofenig, "Secure Telephone Identity Problem Statement", Internet draft (work in progress), draft-ietf-stir-problem-statement-05.txt, May 2014.

[I-D.ietf-stir-threats]

Peterson, J., "Secure Telephone Identity Threat Model", Internet draft (work in progress), draft-ietf-stir-threats-03.txt, June 2014.

[I-D.ietf-stir-rfc4474bis]

Peterson, J., Jennings, C. and E. Rescorla, "Authenticated Identity Management in the Session Initiation Protocol (SIP)", Internet draft (work in progress), draft-ietf-stir-rfc4474bis-01.txt, July 2014.

[I-D.thomson-geopriv-location-dependability]

Thomson, M. and J. Winterbottom, "Digital Signature Methods for Location Dependability", Internet draft (work in progress), draft-thomson-geopriv-location-dependability-07.txt, March 2011.

[EENA]

EENA, "False Emergency Calls", EENA Operations Document, Version 1.1, May 2011, http://www.eena.org/ressource/static/files/2012_05_04-3.1.2.fc_v1.1.pdf

[GPSCounter]

Warner, J. S. and R. G. Johnston, "GPS Spoofing Countermeasures", Los Alamos research paper LAUR-03-6163, December 2003.

[NENA-i2] "08-001 NENA Interim VoIP Architecture for Enhanced 9-1-1 Services (i2)", December 2005.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2818] Rescorla, E., "HTTP over TLS", RFC 2818, May 2000.
- [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.
- [RFC3693] Cuellar, J., Morris, J., Mulligan, D., Peterson, J., and J. Polk, "Geopriv Requirements", RFC 3693, February 2004.
- [RFC3694] Danley, M., Mulligan, D., Morris, J. and J. Peterson, "Threat Analysis of the Geopriv Protocol", RFC 3694, February 2004.
- [RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowitz, "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004.
- [RFC3863] Sugano, H., Fujimoto, S., Klyne, G., Bateman, A., Carr, W. and J. Peterson, "Presence Information Data Format (PIDF)", RFC 3863, August 2004.
- [RFC4119] Peterson, J., "A Presence-based GEOPRIV Location Object Format", RFC 4119, December 2005.
- [RFC4474] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, August 2006.
- [RFC4479] Rosenberg, J., "A Data Model for Presence", RFC 4479, July 2006.
- [RFC4740] Garcia-Martin, M., Belinchon, M., Pallares-Lopez, M., Canales-Valenzuela, C., and K. Tammi, "Diameter Session Initiation Protocol (SIP) Application", RFC 4740, November 2006.
- [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.
- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Level Security (TLS) Protocol Version 1.2", RFC 5246, August 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.
- [RFC5606] Peterson, J., Hardie, T. and J. Morris, "Implications of 'retransmission-allowed' for SIP Location Conveyance", RFC 5606, August 2009.
- [RFC5751] Ramsdell, B. and S. Turner, "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification", RFC 5751, January 2010.
- [RFC5808] Marshall, R., "Requirements for a Location-by-Reference Mechanism", RFC 5808, May 2010.
- [RFC5985] Barnes, M., "HTTP Enabled Location Delivery (HELD)", RFC 5985, September 2010.
- [RFC6280] Barnes, R., et. al, "An Architecture for Location and Location Privacy in Internet Applications", RFC 6280, July 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.
- [RFC6444] Schulzrinne, H., Liess, L., Tschofenig, H., Stark, B., and A. Kuett, "Location Hiding: Problem Statement and Requirements", RFC 6444, January 2012.
- [RFC6753] Winterbottom, J., Tschofenig, H., Schulzrinne, H. and M. Thomson, "A Location Dereference Protocol Using HTTP-Enabled Location Delivery (HELD)", RFC 6753, October 2012.
- [RFC6881] Rosen, B. and J. Polk, "Best Current Practice for Communications Services in Support of Emergency Calling", BCP 181, RFC 6881, March 2013.
- [RFC7090] Schulzrinne, H., Tschofenig, H., Holmberg, C. and M. Patel, "Public Safety Answering Point (PSAP) Callback", RFC 7090, April 2014.
- [RFC7199] Barnes, R., Thomson, M., Winterbottom, J. and H. Tschofenig, "Location Configuration Extensions for Policy Management", RFC 7199, April 2014.

- [SA] "Saudi Arabia - Illegal sale of SIMs blamed for surge in hoax calls", Arab News, May 4, 2010,
http://www.menafn.com/qn_news_story_s.asp?StoryId=1093319384
- [STIR] IETF, "Secure Telephone Identity Revisited (stir) Working Group", <http://datatracker.ietf.org/wg/stir/charter/>, October 2013.
- [Swatting]
"Don't Make the Call: The New Phenomenon of 'Swatting',
Federal Bureau of Investigation, February 4, 2008,
<http://www.fbi.gov/news/stories/2008/february/swatting020408>
- [TASMANIA]
"Emergency services seek SIM-less calls block", ABC News
Online, August 18, 2006,
<http://www.abc.net.au/elections/tas/2006/news/stories/1717956.htm?elections/tas/2006/>
- [UK] "Rapper makes thousands of prank 999 emergency calls to UK police", Digital Journal, June 24, 2010,
<http://www.digitaljournal.com/article/293796?tp=1>

Acknowledgments

We would like to thank the members of the IETF ECRIT working group, including Marc Linsner and Brian Rosen, for their input at IETF 85 that helped get this documented pointed in the right direction. We would also like to thank members of the IETF GEOPRIV WG, including Andrew Newton, Murugaraj Shanmugam, Martin Thomson, Richard Barnes and Matt Lepinski for their feedback to previous versions of this document. Thanks also to Pete Resnick, Adrian Farrel, Alissa Cooper, Bert Wijnen and Meral Shirazipour who provided review comments in IETF last call.

Authors' Addresses

Hannes Tschofenig
Austria

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

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

Phone: +1 212 939 7004
Email: hgs@cs.columbia.edu
URI: <http://www.cs.columbia.edu>

Bernard Aboba
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
US

Email: bernard_aboba@hotmail.com

ECRIT
Internet-Draft
Intended status: Standards Track
Expires: February 13, 2015

H. Schulzrinne
Columbia University
S. McCann
Research in Motion UK Ltd
G. Bajko

H. Tschofenig

D. Kroeselberg
Siemens
August 12, 2014

Extensions to the Emergency Services Architecture for dealing with
Unauthenticated and Unauthorized Devices
draft-ietf-ecrit-unauthenticated-access-10.txt

Abstract

This document provides a problem statement, introduces terminology and describes an extension for the base IETF emergency services architecture to address cases where an emergency caller is not authenticated, has no identifiable service provider, or has no remaining credit with which to pay for access to the network.

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 February 13, 2015.

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	2
2. Terminology	5
3. Use Case Categories	5
4. ZBP Considerations	11
5. NASP Considerations	11
5.1. End Host Profile	14
5.1.1. LoST Server Discovery	14
5.1.2. ESRP Discovery	14
5.1.3. Location Determination and Location Configuration	14
5.1.4. Emergency Call Identification	14
5.1.5. SIP Emergency Call Signaling	14
5.1.6. Media	15
5.1.7. Testing	15
5.2. IAP/ISP Profile	15
5.2.1. ESRP Discovery	15
5.2.2. Location Determination and Location Configuration	15
5.3. ESRP Profile	15
5.3.1. Emergency Call Routing	15
5.3.2. Emergency Call Identification	15
5.3.3. SIP Emergency Call Signaling	16
6. Lower Layer Considerations for NAA Case	16
6.1. Link Layer Emergency Indication	17
6.2. Securing Network Attachment in NAA Cases	18
7. Security Considerations	19
8. Acknowledgments	20
9. IANA Considerations	21
10. References	21
10.1. Normative References	21
10.2. Informative References	22
Authors' Addresses	23

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 older technology. New devices and services are being made available that could be used to make a request for help, those devices are not traditional telephones, and users are increasingly expecting them to be used to place emergency calls.

Roughly speaking, the IETF emergency services architecture (see [RFC6881] and [RFC6443]) divides responsibility for handling emergency calls among the access network (ISP); the application service provider (ASP), which may be a VoIP service provider (VSP); and the provider of emergency signaling services, the emergency service network (ESN). The access network may provide location information to end systems, but does not have to provide any ASP signaling functionality. The emergency caller can reach the ESN either directly or through the ASP's outbound proxy. Any of the three parties can provide the mapping from location to PSAP URI by offering LoST [RFC5222] services.

In general, a set of automated configuration mechanisms allows a device to function in a variety of architectures, without the user being aware of the details on who provides location, mapping services or call routing services. However, if emergency calling is to be supported when the calling device lacks access network authorization or does not have an ASP, one or more of the providers may need to provide additional services and functions.

In all cases, the end device has to be able to perform a LoST lookup and otherwise conduct the emergency call in the same manner as when the three exceptional conditions discussed below do not apply.

We distinguish among three conditions:

No Access Authentication (NAA): In the NAA case, the emergency caller does not possess valid credentials for the access network. This includes the case where the access network allows pay-per-use, as is common for wireless hotspots, but there is insufficient time to enter credit card details and other registration information required for access. It also covers all cases where either no credentials are available at all, or the available credentials do not work for the given IAP/ISP. As a result, the NAA case basically combines the below NASP and ZBP cases, but at the IAP/ISP level. Support for emergency call handling in the NAA case is subject to the local policy of the ISP. Such policy may vary substantially between ISPs and typically depends on external factors that are not under the ISP control.

No ASP (NASP): The caller does not have an ASP at the time of the call. This can occur either in case the caller does not possess any valid subscription for a reachable ASP, or in case none of the ASPs where the caller owns a valid subscription is reachable through the ISP.

Note: The interoperability need is increased with this scenario since the client software used by the emergency caller must be compatible with the protocols and extensions deployed by the ESN.

Zero-balance ASP (ZBP): In the case of zero-balance ASP, the ASP can authenticate the caller, but the caller is not authorized to use ASP services, e.g., because the contract has expired or the prepaid account for the customer has been depleted.

These three cases are not mutually exclusive. A caller in need of help may, for example, be in a NAA and NASP situation, as explained in more detail in Figure 1. Depending on local policy and regulations, it may not be possible to place emergency calls in the NAA case. Unless local regulations require user identification, it should always be possible to place calls in the NASP case, with minimal impact on the ISP. Unless the ESN requires that all calls traverse a known set of VSPs, it is technically possible to let a caller place an emergency call in the ZBP case. We discuss each case in more details in Section 3.

As mentioned in the abstract some of the functionality provided in this document is already available in the PSTN. Consequently, there is real-world experience available and not all of it is positive. For example, the functionality of SIM-less calls in today's cellular system has lead to a fair amount of hoax or test calls in certain countries. This causes overload situations at PSAPs, which is considered harmful to the overall availability and reliability of emergency services.

As an example, Federal Office of Communications (OFCOM, Switzerland) provided statistics about emergency (112) calls in Switzerland from Jan. 1997 to Nov. 2001. Switzerland did not offer SIM-less emergency calls except for almost a month in July 2000 where a significant increase in hoax and test calls was reported. As a consequence, the functionality was disabled again. More details can be found in the panel presentations of the 3rd SDO Emergency Services Workshop [esw07].

2. Terminology

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

This document reuses terminology from [RFC5687] and [RFC5012], namely Internet Access Provider (IAP), Internet Service Provider (ISP), Application Service Provider (ASP), Voice Service Provider (VSP), Emergency Service Routing Proxy (ESRP), Public Safety Answering Point (PSAP), Location Configuration Server (LCS), (emergency) service dial string, and (emergency) service identifier.

3. Use Case Categories

On a very high-level, the steps to be performed by an end host that is not attached to the network and the user starting to make an emergency call are the following:

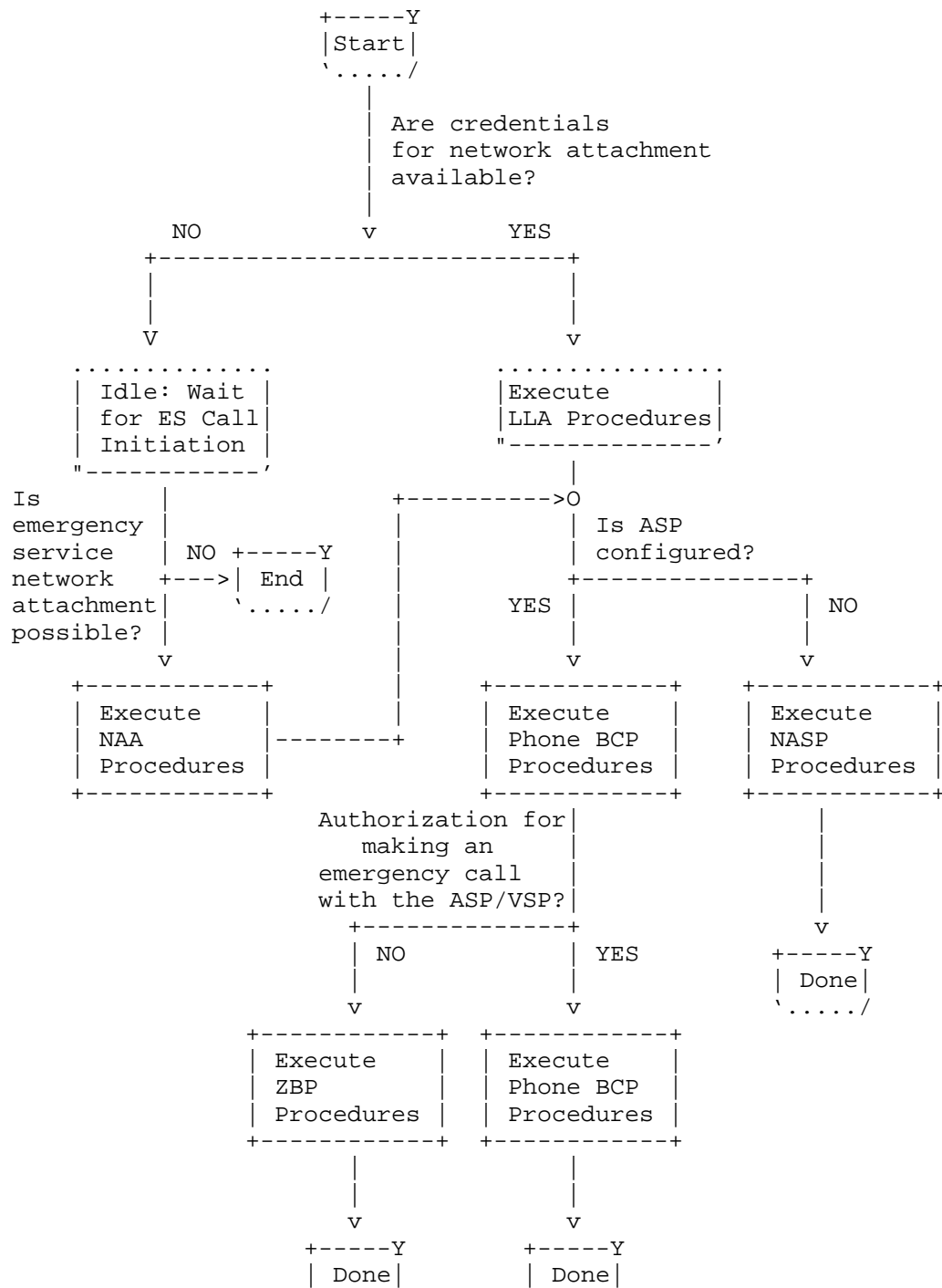
Link Layer Attachment: Some networks have added support for unauthenticated emergency access, some other type of networks advertise these capabilities using layer beacons. The end host learns about these unauthenticated emergency services capabilities either from the link layer type or from advertisement.

The end host uses the link layer specific network attachment procedures defined for unauthenticated network access in order to get access to the network.

Pre-Emergency Service Configuration: When the link layer network attachment procedure is completed the end host learns basic configuration information using DHCP from the ISP. The end host uses a Location Configuration Protocol (LCP) to retrieve location information. Subsequently, the LoST protocol [RFC5222] is used to learn the relevant emergency numbers, and to obtain the PSAP URI applicable for that location.

Emergency Call: In case of need for help, a user dials an emergency number and the SIP UA initiates the emergency call procedures by communicating with the PSAP.

Figure 1 compiles the basic logic taking place during network entry for requesting an emergency service and shows the interrelation between the three conditions described in the above section.



\...../ \...../

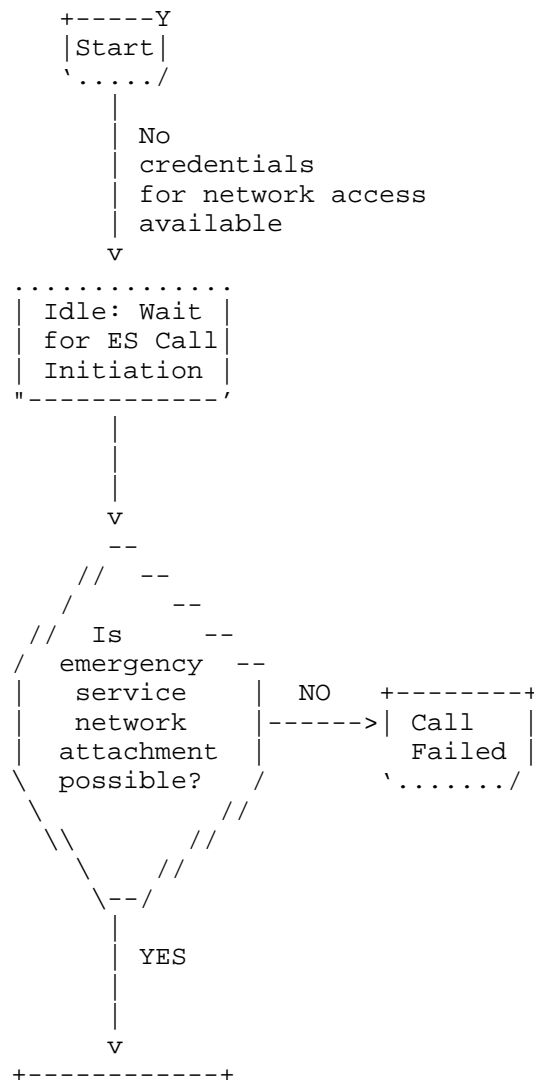
Abbreviations:

LLA: Link Layer Attachment

ES: Emergency Services

Figure 1: Flow Diagram: NAA, ZBP, and NSAP Scenarios.

The diagrams below highlight the most important steps for the three cases.



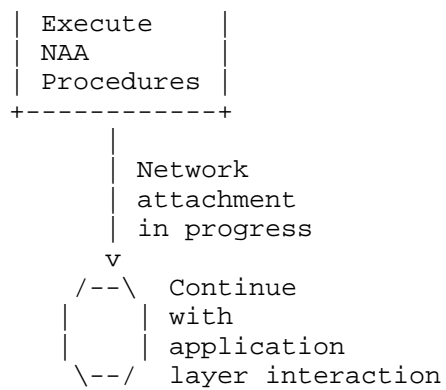
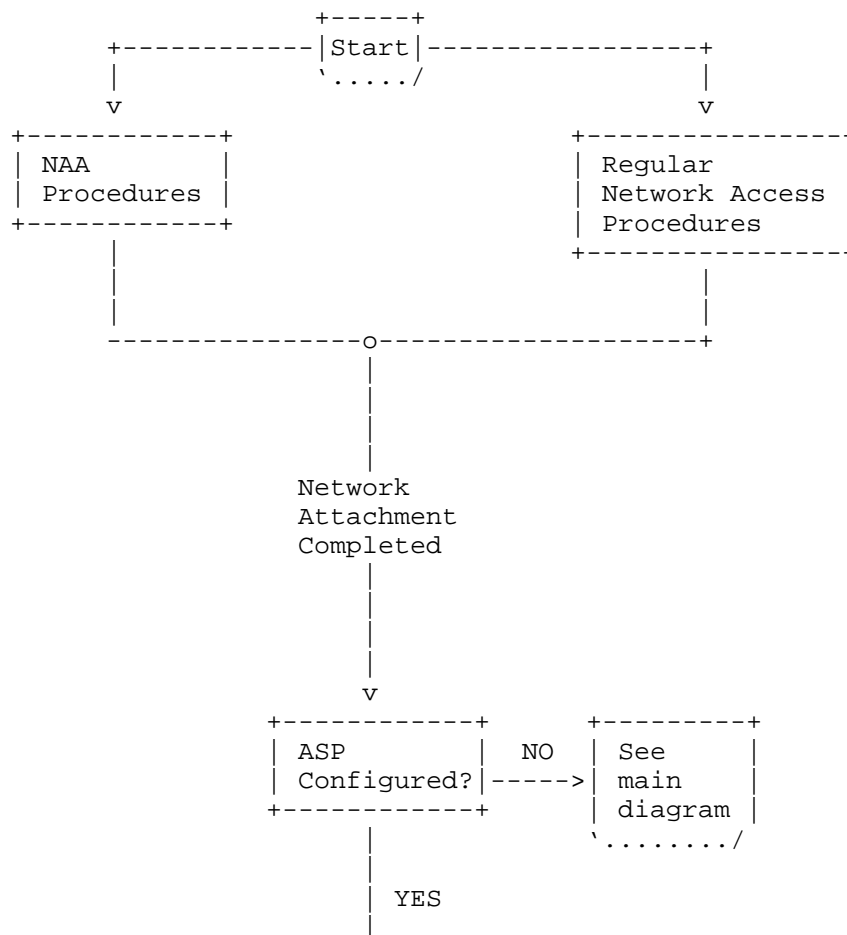


Figure 2: Flow Diagram: NAA Scenario.



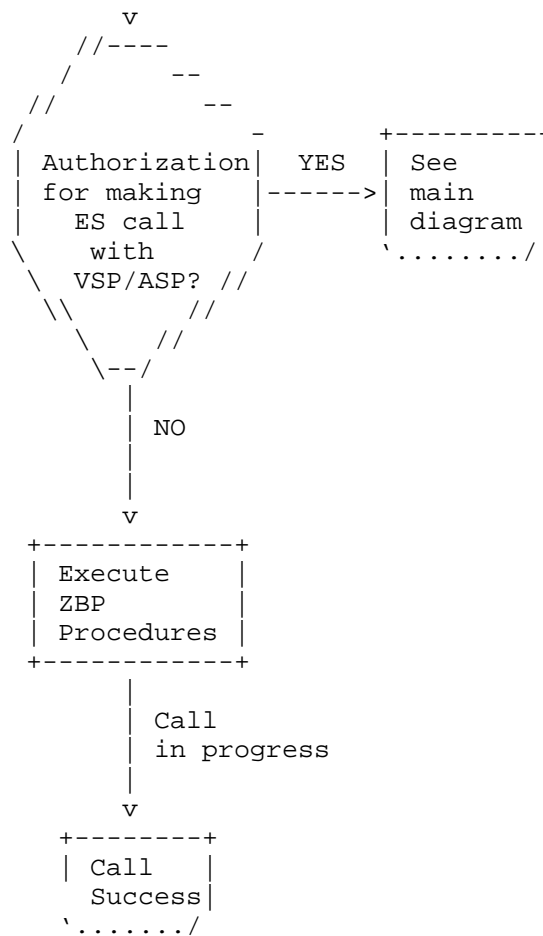
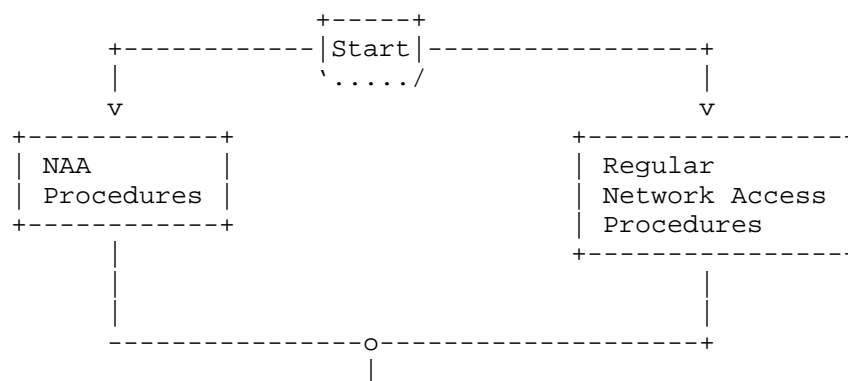


Figure 3: Flow Diagram: ZBP Scenario.



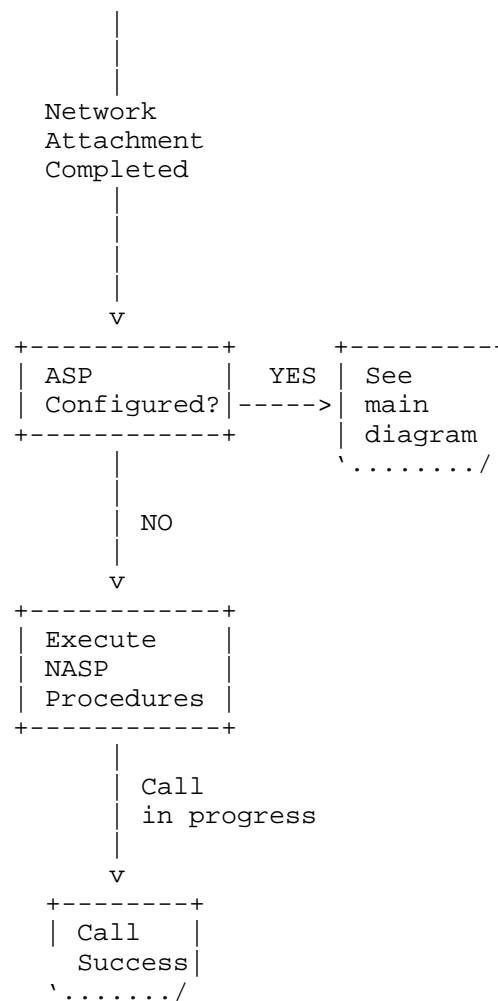


Figure 4: Flow Diagram: NASP Scenario.

The "No Access Authentication (NAA)" procedures are described in Section 6. The "Zero-balance ASP (ZBP)" procedures are described in Section 4. The "No ASP (NASP)" procedures are described in Section 5. The Phone BCP procedures are described in [RFC6881]. The "Link Layer Attachment (LLA)" procedures are not described in this document since they are specific to the link layer technology in use.

4. ZBP Considerations

ZBP includes all cases where a subscriber is known to an ASP, but lacks the necessary authorization to access regular ASP services. Example ZBP cases include empty prepaid accounts, barred accounts, roaming and mobility restrictions, or any other conditions set by ASP policy.

Local regulation might demand that emergency calls cannot proceed without successful service authorization. In regulatory regimes, however, it may be possible to allow emergency calls to continue despite authorization failures. To distinguish an emergency call from a regular call an ASP can identify emergency sessions by inspecting the service URN [RFC5031] used in call setup. The ZBP case therefore only affects the ASP.

Permitting a call despite authorization failures could present an opportunity for abuse. The ASP may choose to verify the destination of the emergency calls and to only permit calls to certain, pre-configured entities (e.g., to local PSAPs). Section 7 discusses this topic in more detail.

An ASP without a regulatory requirement to authorize emergency calls can deny emergency call setup. Where an ASP does not authorize an emergency call, the caller may be able to fall back to NASP procedures.

5. NASP Considerations

To start the description we consider the sequence of steps that are executed in an emergency call based on Figure 5.

- o As an initial step the devices attaches to the network as shown in step (1). This step is outside the scope of this section.
- o When the link layer network attachment procedure is completed the end host learns basic IP configuration information using DHCP from the ISP, as shown in step (2).
- o When the IP address configuration is completed then the end host starts an interaction with the discovered Location Configuration Server at the ISP, as shown in step (3). The ISP may in certain deployments need to interact with the IAP. This protocol exchange is shown in step (4).
- o Once location information is obtained the end host triggers the LoST protocol to obtain the address of the ESRP/PSAP. This step is shown in (5).

- o In step (6), the SIP UA initiates a SIP INVITE towards the indicated ESRP. The INVITE message contains all the necessary parameters required by Section 5.1.5.
- o The ESRP receives the INVITE and processes it according to the description in Section 5.3.3.
- o The ESRP routes the call to the PSAP, as shown in (8), potentially interacting with a LoST server first to determine the route.
- o The PSAP evaluates the initial INVITE and aims to complete the call setup.
- o Finally, when the call setup is completed media traffic can be exchanged between the PSAP and the SIP UA.

For editorial reasons the end-to-end SIP and media exchange between the PSAP and SIP UA are not shown in Figure 5.

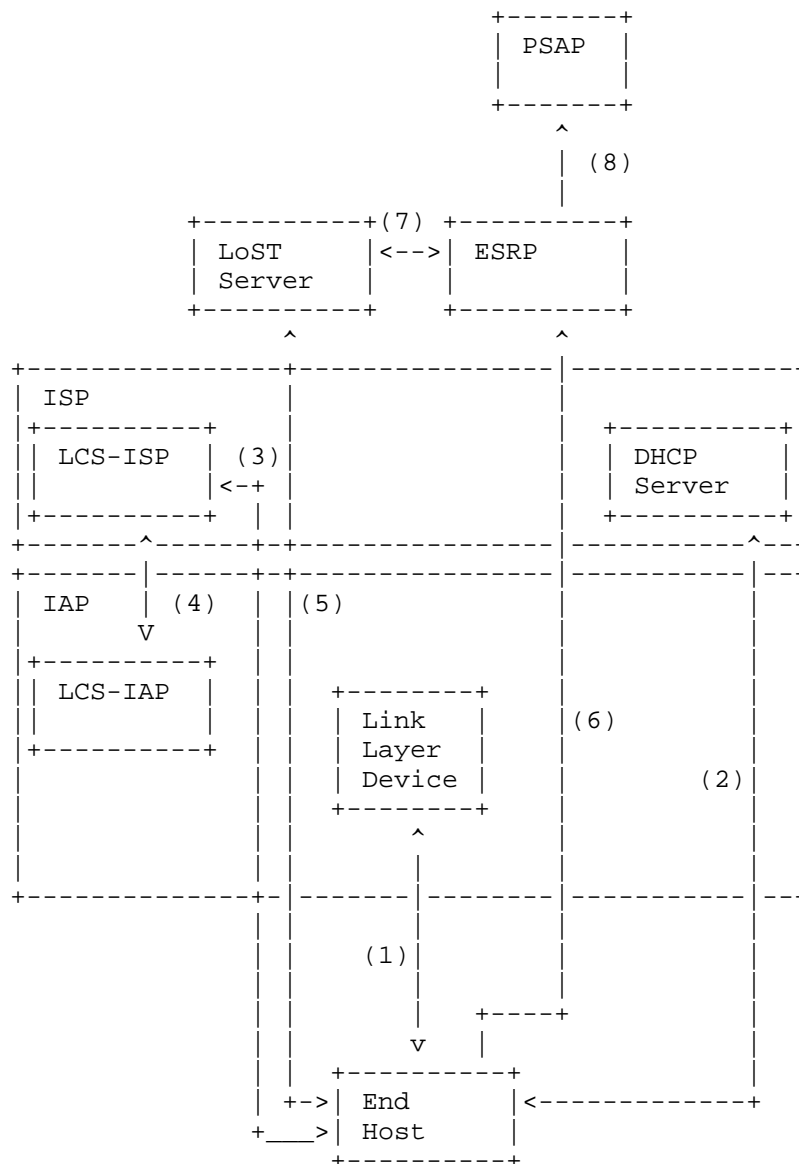


Figure 5: Architectural Overview

Note: Figure 5 does not indicate who operates the ESRP and the LoST server. Various deployment options exist.

5.1. End Host Profile

5.1.1. LoST Server Discovery

The end host MUST discover a LoST server [RFC5222] using DHCP [RFC5223] unless a LoST server has been provisioned using other means.

5.1.2. ESRP Discovery

The end host MUST discover the ESRP using the LoST protocol [RFC5222] unless a ESRP has been provisioned using other means.

5.1.3. Location Determination and Location Configuration

The end host MUST support location acquisition and the LCPs described in Section 6.5 of [RFC6881]. The description in Section 6.5 and 6.6 of [RFC6881] regarding the interaction between the device and the LIS applies to this document.

The SIP UA in the end host MUST attach available location information in a PIDF-LO [RFC4119] when making an emergency call. When constructing the PIDF-LO the guidelines in PIDF-LO profile [RFC5491] MUST be followed. For civic location information the format defined in [RFC5139] MUST be supported.

5.1.4. Emergency Call Identification

To determine which calls are emergency calls, some entity needs to map a user entered dialstring into this URN scheme. A user may "dial" 1-1-2, 9-1-1, etc., but the call would be sent to urn:service:sos. This mapping SHOULD be performed at the endpoint device.

End hosts MUST use the Service URN mechanism [RFC5031] to mark calls as emergency calls for their home emergency dial string.

5.1.5. SIP Emergency Call Signaling

SIP signaling capabilities [RFC3261] are REQUIRED for end hosts.

The initial SIP signaling method is an INVITE. The SIP INVITE request MUST be constructed according to the requirements in Section 9.2 [RFC6881].

Regarding callback behavior SIP UAs SHOULD place a globally routable URI in a Contact: header.

5.1.6. Media

End points MUST comply with the media requirements for end points placing an emergency call found in Section 14 of [RFC6881].

5.1.7. Testing

The description in Section 15 of [RFC6881] is fully applicable to this document.

5.2. IAP/ISP Profile

5.2.1. ESRP Discovery

An ISP MUST provision a DHCP server with information about LoST servers [RFC5223]. An ISP operator may choose to deploy a LoST server or to outsource it to other parties.

5.2.2. Location Determination and Location Configuration

The ISP is responsible for location determination and exposes this information to the end points via location configuration protocols. The considerations described in [RFC6444] are applicable to this document.

The ISP MUST support one of the LCPs described in Section 6.5 of [RFC6881]. The description in Section 6.5 and 6.6 of [RFC6881] regarding the interaction between the end device and the LIS applies to this document.

The interaction between the LIS at the ISP and the IAP is often proprietary but the description in [I-D.winterbottom-geopriv-lis2lis-req] may be relevant to the reader.

5.3. ESRP Profile

5.3.1. Emergency Call Routing

The ESRP continues to route the emergency call to the PSAP responsible for the physical location of the end host. This may require further interactions with LoST servers but depends on the specific deployment.

5.3.2. Emergency Call Identification

The ESRP MUST understand the Service URN mechanism [RFC5031] (i.e., the 'urn:service:sos' tree).

5.3.3. SIP Emergency Call Signaling

SIP signaling capabilities [RFC3261] are REQUIRED for the ESRP. The ESRP MUST process the messages sent by the client, according to Section 5.1.5.

Furthermore, if a PSAP wants to support NASP calls, then it MUST NOT restrict incoming calls to a particular set of ASPs.

6. Lower Layer Considerations for NAA Case

Some networks have added support for unauthenticated emergency access, some other type of networks advertise these capabilities using layer beacons. The end host learns about these unauthenticated emergency services capabilities either from the link layer type or from advertisement.

It is important to highlight that the NAA case is inherently a layer 2 problem, and the general form of the solution is to provide an "emergency only" access type, with appropriate limits/monitoring to prevent abuse. The described mechanisms are informative in nature since the relationship to the IETF emergency services architecture is only indirect, namely via some protocols developed within the IETF (e.g., EAP and EAP methods) that require extensions to support this functionality.

This section discusses different methods to indicate an emergency service request as part of network attachment. It provides some general considerations and recommendations that are not specific to the access technology.

To perform network attachment and get access to the resources provided by an IAP/ISP, the end host uses access technology specific network attachment procedures, including for example network detection and selection, authentication, and authorization. For initial network attachment of an emergency service requester, the method of how the emergency indication is given to the IAP/ISP is specific to the access technology. However, a number of general approaches can be identified:

Link layer emergency indication: The end host provides an indication, e.g., an emergency parameter or flag, as part of the link layer signaling for initial network attachment. Examples include an emergency bit signalled in the IEEE 802.16-2009 wireless link. In IEEE 802.11 WLAN, an emergency support indicator allows the station (i.e., end host in this context) to download before association a Network Access Identifier (NAI),

which it can use to request server side authentication only for an 802.1x network.

Higher-layer emergency indication: Typically, emergency indication is provided in the network access authentication procedure. The emergency caller's end host provides an indication as part of the access authentication exchanges. Authentication via the Extensible Authentication Protocol (EAP) [RFC3748] is of particular relevance here. Examples are the EAP NAI decoration used in WiMAX networks and modification of the authentication exchange in IEEE 802.11. [nwgstg3].

6.1. Link Layer Emergency Indication

In general, link layer emergency indications provide good integration into the actual network access procedure regarding the enabling of means to recognize and prioritize an emergency service request from an end host at a very early stage of the network attachment procedure. However, support in end hosts for such methods cannot be considered to be commonly available.

No general recommendations are given in the scope of this memo due to the following reasons:

- o Dependency on the specific access technology.
- o Dependency on the specific access network architecture. Access authorization and policy decisions typically happen at a different layers of the protocol stack and in different entities than those terminating the link-layer signaling. As a result, link layer indications need to be distributed and translated between the different involved protocol layers and entities. Appropriate methods are specific to the actual architecture of the IAP/ISP network.
- o An advantage of combining emergency indications with the actual network attachment procedure performing authentication and authorization is the fact that the emergency indication can directly be taken into account in the authentication and authorization server that owns the policy for granting access to the network resources. As a result, there is no direct dependency on the access network architecture that otherwise would need to take care of merging link-layer indications into the AA and policy decision process.

- o EAP signaling happens at a relatively early stage of network attachment, so it is likely to match most requirements for prioritization of emergency signaling. However, it does not cover early stages of link layer activity in the network attachment process. Possible conflicts may arise e.g. in case of MAC-based filtering in entities terminating the link-layer signaling in the network (like a base station). In normal operation, EAP related information will only be recognized in the NAS. Any entity residing between end host and NAS should not be expected to understand/parse EAP messages.
- o An emergency indication can be given by forming a specific NAI that is used as the identity in EAP based authentication for network entry.

6.2. Securing Network Attachment in NAA Cases

For network attachment in NAA cases, it may make sense to secure the link-layer connection between the device and the IAP/ISP. This especially holds for wireless access with examples being IEEE 802.11 or IEEE 802.16 based access. The latter even mandates secured communication across the wireless link for all IAP/ISP networks based on [nwgstg3].

Therefore, for network attachment that is by default based on EAP authentication it is desirable also for NAA network attachment to use a key-generating EAP method (that provides an MSK key to the authenticator to bootstrap further key derivation for protecting the wireless link).

The following approaches to match the above can be identified:

1) Server-only Authentication:

The device of the emergency service requester performs an EAP method with the IAP/ISP EAP server that performs server side authentication only. An example for this is EAP-TLS [RFC5216]. This provides a certain level of assurance about the IAP/ISP to the device user. It requires the device to be provisioned with appropriate trusted root certificates to be able to verify the server certificate of the EAP server (unless this step is explicitly skipped in the device in case of an emergency service request). This method is used to provide access of devices without existing credentials to an 802.1x network. The details are incorporated into the not yet published 802.11-2011 specification.

2) Null Authentication:

In one case (e.g., WiMAX) an EAP method is performed. However, no credentials specific to either the server or the device or subscription are used as part of the authentication exchange. An example for this would be an EAP-TLS exchange with using the TLS_DH_anon (anonymous) ciphersuite. Alternatively, a publicly available static key for emergency access could be used. In the latter case, the device would need to be provisioned with the appropriate emergency key for the IAP/ISP in advance. In another case (e.g., IEEE 802.11), no EAP method is used, so that empty frames are transported during the over the air IEEE 802.1X exchange. In this case the authentication state machine completes with no cryptographic keys being exchanged.

3) Device Authentication:

This case extends the server-only authentication case. If the device is configured with a device certificate and the IAP/ISP EAP server can rely on a trusted root allowing the EAP server to verify the device certificate, at least the device identity (e.g., the MAC address) can be authenticated by the IAP/ISP in NAA cases. An example for this are WiMAX devices that are shipped with device certificates issued under the global WiMAX device public-key infrastructure. To perform unauthenticated emergency calls, if allowed by the IAP/ISP, such devices perform EAP-TLS based network attachment with client authentication based on the device certificate.

7. Security Considerations

The security threats discussed in [RFC5069] are applicable to this document.

There are a couple of new vulnerabilities raised with unauthenticated emergency services in NASP/NAA cases since the PSAP operator will typically not possess any identity information about the emergency caller via the signaling path itself. In countries where this functionality is used for GSM networks today this has lead to a significant amount of misuse.

In the context of NAA, the IAP and the ISP will probably want to make sure that the claimed emergency caller indeed performs an emergency call rather than using the network for other purposes, and thereby acting fraudulent by skipping any authentication, authorization and accounting procedures. By restricting access of the unauthenticated emergency caller to the LoST server and the PSAP URI, traffic can be

restricted only to emergency calls. This can be accomplished with traffic separation. The details, however, e.g. for using filtering, depend on the deployed ISP architecture and are beyond the scope of this document.

We only illustrate a possible model. If the ISP runs its own (caching) LoST server, the ISP would maintain an access control list populated with IP-address information obtained from LoST responses (in the mappings). These URIs would either be URIs for contacting further LoST servers or PSAP URIs. It may be necessary to translate domain names returned in LoST responses to IP addresses. Since the media destination addresses are not predictable, the ISP also has to provide a SIP outbound proxy so that it can determine the media addresses and add those to the filter list.

For the ZBP case the additional aspect of fraud has to be considered. Unless the emergency call traverses a PSTN gateway or the ASP charges for IP-to-IP calls, there is little potential for fraud. If the ASP also operates the LoST server, the outbound proxy MAY restrict outbound calls to the SIP URIs returned by the LoST server. It is NOT RECOMMENDED to rely on a fixed list of SIP URIs, as that list may change.

RFC 6280 [RFC6280] discusses security vulnerabilities that are caused by an adversary faking location information and thereby lying about the actual location of the emergency caller. These threats may be less problematic in the context of unauthenticated emergency when location information can be verified by the ISP to fall within a specific geographical area.

8. Acknowledgments

Parts of this document are derived from [RFC6881]. Participants of the 2nd and 3rd SDO Emergency Services Workshop provided helpful input.

We would like to thank Richard Barnes, Brian Rosen, James Polk, Marc Linsner, and Martin Thomson for their feedback at the IETF#80 ECRIT meeting.

Furthermore, we would like to thank Martin Thomson and Bernard Aboba for their detailed document review in preparation of the 81st IETF meeting. Alexey Melnikov was the General Area (Gen-Art) reviewer. A number of changes to the document had been made in response to the AD review by Richard Barnes.

We would also like to thank review comments from various IESG members, including Stephen Farrell, Barry Leiba, Pete Resnick, Spencer Dawkins, Joel Jaeggli, and Ted Lemon.

9. IANA Considerations

This document does not require actions by IANA.

10. References

10.1. Normative References

- [RFC5031] Schulzrinne, H., "A Uniform Resource Name (URN) for Emergency and Other Well-Known Services", RFC 5031, January 2008.
- [RFC4119] Peterson, J., "A Presence-based GEOPRIV Location Object Format", RFC 4119, December 2005.
- [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.
- [RFC5139] Thomson, M. and J. Winterbottom, "Revised Civic Location Format for Presence Information Data Format Location Object (PIDF-LO)", RFC 5139, February 2008.
- [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.
- [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", BCP 181, RFC 6881, March 2013.
- [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.

10.2. Informative References

- [RFC5687] Tschofenig, H. and H. Schulzrinne, "GEOPRIV Layer 7 Location Configuration Protocol: Problem Statement and Requirements", RFC 5687, March 2010.
- [RFC6443] Rosen, B., Schulzrinne, H., Polk, J., and A. Newton, "Framework for Emergency Calling Using Internet Multimedia", RFC 6443, December 2011.
- [RFC5012] Schulzrinne, H. and R. Marshall, "Requirements for Emergency Context Resolution with Internet Technologies", RFC 5012, January 2008.
- [RFC6444] Schulzrinne, H., Liess, L., Tschofenig, H., Stark, B., and A. Kuett, "Location Hiding: Problem Statement and Requirements", RFC 6444, January 2012.
- [I-D.winterbottom-geopriv-lis2lis-req] Winterbottom, J. and S. Norreys, "LIS to LIS Protocol Requirements", draft-winterbottom-geopriv-lis2lis-req-01 (work in progress), November 2007.
- [RFC5069] Taylor, T., Tschofenig, H., Schulzrinne, H., and M. Shanmugam, "Security Threats and Requirements for Emergency Call Marking and Mapping", RFC 5069, January 2008.
- [RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J., and H. Levkowetz, "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004.
- [RFC5216] Simon, D., Aboba, B., and R. Hurst, "The EAP-TLS Authentication Protocol", RFC 5216, March 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.
- [esw07] "3rd SDO Emergency Services Workshop, <http://www.emergency-services-coordination.info/2007Nov/>", October 30th - November 1st 2007.

[nwgstg3] "WiMAX Forum WMF-T33-001-R015V01, WiMAX Network Architecture Stage-3
http://www.wimaxforum.org/sites/wimaxforum.org/files/technical_document/2009/09/DRAFT-T33-001-R015v01-O_Network-Stage3-Base.pdf", September 2009.

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
URI: <http://www.cs.columbia.edu>

Stephen McCann
Research in Motion UK Ltd
200 Bath Road
Slough, Berks SL1 3XE
UK

Phone: +44 1753 667099
Email: smccann@rim.com
URI: <http://www.rim.com>

Gabor Bajko

Email: gaborbajko@gmail.com

Hannes Tschofenig
Hall in Tirol 6060
Austria

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

Dirk Kroeselberg
Siemens
Germany

Email: dirk.kroeselberg@siemens.com