ECRIT	B. Rosen
Internet-Draft	NeuStar, Inc.
Intended status: Experimental Protocol	H. Schulzrinne
Expires: January 12, 2012	Columbia U.
	H. Tschofenig
	Nokia Siemens Networks
	July 11, 2011

Common Alerting Protocol (CAP) based Emergency Alerts using the Session Initiation Protocol (SIP)

draft-ietf-ecrit-data-only-ea-02.txt

Abstract

The Common Alerting Protocol (CAP) is a document format for exchanging emergency alerts and public warnings. CAP is mainly used for conveying alerts and warnings between authorities and from authorities to citizen/individuals. This document describes how devices use CAP to issue emergency alerts.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

The Common Alerting Protocol (CAP) [cap] is an XML document format for exchanging emergency alerts and public warnings. CAP is mainly used for conveying alerts and warnings between authorities and from authorities to citizen/individuals. This document describes how data-only emergency calls are able to utilize the same CAP document format.

Emergency alerts containing data are similar to regular emergency calls in the sense that they require emergency call routing functionality and may even have the same location requirements. On the other hand, the communication interaction may occur without establishment of a voice or video channel.
Data-only emergency alerts are similar to regular emergency calls in the sense that they require emergency call routing functionality and may even have the same location requirements. On the other hand, the initial communication interaction will not lead to the establishment of a voice or video channel.
Based on the deployment experience with non-IP based systems, two major deployment scenarios are envisaged:

1. Emergency alerts containing only data are targeted to a recipient responsible for evaluating the next steps, which could include:

    1. Sending an alert containing only data toward a Public Safety Answering Point (PSAP);

    2. Establishing an emergency call with a PSAP that could include audio/video as well as data

2. Emergency alerts targeted to a Service URN 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 and a PIDF-LO payload in a SIP MESSAGE) or could be included along with establishment of an audio/video channel (e.g. SIP INVITE)

We describe these two cases in more detail in Section 3.

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 utilizes terminology introduced in [I-D.ietf-atoca-requirements]. In particular, the terms for author, originator, receiver and recipient, are relevant for this document. The originator and the receiver are SIP-based entities while the author and the recipient are entities that relate to the alert message delivery, when this is relevant for the communication.

3. Architectural Overview

This section illustrates two envisioned usage modes; targeted and location-based emergency alert routing. Figure 1 shows a deployment variant where a sensor, as the author and originator of the alert, is pre-configured (using techniques outside the scope of this document) to

issue an alert to a receiver or an aggregator, a special form of mediator, that processes these messages and performs whatever steps are necessary to appropriately react on 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.

```
 +------------+             +------------+
 | Sensor     |             | Aggregator |
 |            |             |            |
 +---+--------+             +------+-----+
     |                             |
  Sensors                          |
  trigger                          |
  emergency                        |
  alert                            |
     |        MESSAGE with CAP     |
     |---------------------------->|
     |                             |
     |                         Aggregator
     |                         processes
     |                         emergency
     |                         alert
     |        200 (OK)             |
     |<----------------------------|
     |                             |
     |                             |
```

In Figure 2 a scenario is shown whereby the alert is routed using location information and the Service URN. An emergency services routing proxy (ESRP) may use LoST 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 call taker. In the generic case, there is very likely no prior relationship between the originator and the receiver, e.g. PSAP. A PSAP, for example, is likely to receive and accept alerts from entities it cannot authorize. This scenario corresponds more to the classical emergency services use case and the description in [I-D.ietf-ecrit-phonebcp] is applicable.

```
                        +-----------+                  +----------+
   +--------+           | ESRP      |                  | PSAP     |
   | Sensor |           |           |                  |          |
   +---+----+           +---+-------+                  +---+------+
       |                    |                              |
    Sensors                 |                              |
    trigger                 |                              |
    emergency               |                              |
    alert                   |                              |
       |                    |                              |
       |                    |                              |
       | MESSAGE with CAP   |                              |
       | (including Service URN,                           |
       | such as urn:service:sos)                          |
       |------------------->|                              |
       |                    |                              |
       |              ESRP performs                        |
       |              emergency alert                      |
       |              routing                              |
       |                    |  MESSAGE with CAP            |
       |                    |  (including identity info)   |
       |                    |----------------------------->|
       |                    |                              |
       |                    |                           PSAP
       |                    |                           processes
       |                    |                           emergency
       |                    |                           alert
       |                    |        200 (OK)              |
       |                    |<-----------------------------|
       |                    |                              |
       |  200 (OK)          |                              |
       |<-------------------|                              |
       |                    |                              |
       |                    |                              |
```

4. Protocol Specification

4.1. CAP Transport

Since alerts structured via CAP require a "push" medium. The following SIP requests MAY carry the CAP payload defined in this document: INVITE [RFC3261], UPDATE [RFC3311], MESSAGE [RFC3428], INFO [RFC6086], NOTIFY [RFC3265], and PUBLISH [RFC3903]. The MIME type is set to 'application/cap+xml'.
If the server does not support the functionality required to fulfill the request then a 501 Not Implemented MUST be returned by RFC 3261 [RFC3261]. This is the appropriate response when a UAS does not

recognize the request method and is not capable of supporting it for any user.
The 415 Unsupported Media Type error MUST be returned by RFC 3261 [RFC3261] if the 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 field, 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 the usage with SIP the following additional requirements are imposed:

sender: A few sub-categories for putting a value in the <sender> element have to be considered:

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: In case that the alert was created by a non-SIP based entity and the identity of this original sender wants to be preserved then this identity MUST be placed into the <sender> element. In this category the it is not useful to be explicit about the author of the alert. The specific type of identity being used will depends on the technology being used by the original originator.

Author indication relevant: In case the author is different from the actual originator of the message and this distinction wants to be preserved then the <sender> element MUST NOT contain the SIP URI.

incidents: The <incidents> element MUST be present whenever there is a possibility that alert information needs to be updated. The initial message will then contain an incident identifier carried in the <incidents> element. 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 may not be present.

scope: The value of the <scope> element MUST be set to "Private" as 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 headers. Populating information twice into different parts of the message may lead to inconsistency.

parameter: The <parameter> element MAY contain additional information specific to the sensor.

area: It is RECOMMENDED to omit this element when constructing a message. In case that the CAP message already contained an <area> element then the specified location information MUST be copied into the PIDF-LO structure of the 'geolocation' header.

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 due to its included alert content, indicating that it was malformed or not satisfactory for the recipient's purpose.
A SIP intermediary can also reject an alert it receives from a UA when it understands that the provided alert is malformed.
Section 5.2 describes a 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 only appropriate to generate a 425 response when the responding entity has no other information in the request that are usable by the responder.
A 425 response code MUST NOT be sent in response to a request that lacks an alert message entirely, as the user agent in that case may not support this extension at all.
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 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 3261)
AlertMsg-Error        = "AlertMsg-Error" HCOLON
                             ErrorValue
ErrorValue       =  error-code
                            *(SEMI error-params)
error-code      = 1*3DIGIT
error-params    = error-code-text
                           / generic-param ; from RFC3261
error-code-text = "code" EQUAL quoted-string ; from RFC3261
```

HCOLON, SEMI, and EQUAL are defined in RFC3261 [RFC3261]. DIGIT is defined in RFC5234 [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 understandable. The text string are OPTIONAL, but RECOMMENDED for human readability, similar to the string phrase used for SIP response codes. That said, the strings are complete enough for rendering to the user, if so desired. 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. Similar to how RFC 3261 specifies, there MUST NOT be more than one string per error code.

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

If, on the other hand, the PSAP cannot accept the MESSAGE 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 MESSAGE may also sends a 425 with a AlertMsg-Error code.

This document defines an initial list of error code ranges for any SIP response, including provisional responses (other than 100 Trying) and the new 425 response. There MUST be no more than one AlertMsg-Error code in a SIP response.

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

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

AlertMsg-Error: 102 ; code="Not enough information to determine the purpose of the alert"
AlertMsg-Error: 103 ; code="Alert Payload was corrupted"
Additionally, if an LR 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. Example

Figure 4 shows a CAP document indicating a BURLARY alert issued by a sensor with the identity 'sensor1@domain.com'. The location of the sensor can be obtained from the attached location information provided via the 'geolocation' header contained in the SIP MESSAGE structure. Additionally, the sensor provided some data long with the alert message using proprietary information elements only to be processed by the receiver, a SIP entity acting as an aggregator. This example reflects the description in Figure 1.

```
 MESSAGE sip:aggregator@domain.com SIP/2.0
 Via: SIP/2.0/TCP sensor1.domain.com;branch=z9hG4bK776sgdkse
 Max-Forwards: 70
 From: sip:sensor1@domain.com;tag=49583
 To: sip:aggregator@domain.com
 Call-ID: asd88asd77a@1.2.3.4
 Geolocation: <cid:abcdef@domain.com>
   ;routing-allowed=yes
 Supported: geolocation
 Accept: application/pidf+xml, application/cap+xml
 CSeq: 1 MESSAGE
 Content-Type: multipart/mixed; boundary=boundary1
 Content-Length: ...

 --boundary1

 Content-Type: cap+xml
 Content-ID: <abcdef2@domain.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@domain.com</sender>
  <sent>2008-11-19T14:57:00-07:00</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@domain.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>32.86726 -97.16054</gml:pos>
                </gml:Point>
             </gml:location>
            </gp:location-info>
            <gp:usage-rules>
              <gbp:retransmission-allowed>false
              </gbp:retransmission-allowed>
              <gbp:retention-expiry>2010-11-14T20:00:00Z
              </gbp:retention-expiry>
            </gp:usage-rules>
            <gp:method>802.11</gp:method>
          </gp:geopriv>
          <dm:timestamp>2010-11-04T20:57:29Z</dm:timestamp>
        </dm:device>
      </presence>
   --boundary1--
```

7. Security Considerations

This section discusses security considerations when SIP user agents issue emergency alerts utilizing CAP. Location specific threats are not unique to this document and are discussed in [I-D.ietf-ecrit-trustworthy-location] and [I-D.ietf-sipcore-location-conveyance]. The ECRIT emergency services architecture [I-D.ietf-ecrit-phonebcp] considers classical individual-to-authority emergency calling and 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 will not depend on the identity of the caller. In case of emergency alerts generated by devices, like sensors, the processing may be different in order to reduce the number of falsely generated emergency alerts. Alerts may get triggered based on certain sensor input that may have been caused by other factors than the actual occurrence of an

alert relevant event. For example, a sensor may simply be malfunctioning. For this purpose not all alert messages are directly sent to a PSAP but are rather 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. These two message routing examples are shown in Figure 1 and in Figure 2.
In any case, for alerts that are initiated by sensors the identity may 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 authorized sensor input will be processed. For this purpose it needs to be ensured that no alert messages from an unknown origin are accepted. 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. These mechanisms can be re-used, such as P-Asserted-Identity [RFC3325] or SIP Identity [RFC4474]. The latter provides a cryptographic assurance while the former relies on a chain of trust model.

2. CAP provides additional security mechanisms and the ability to carry additional information about the sender's identity. Section 3.3.2.1 of [cap] specifies the signing algorithms of CAP documents.

In addition to the desire to perform identity-based access control the classical communication security threats need to be considered, including integrity protection to prevent forgery and 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. These attributes make the CAP document unique for a specific sender and provide time restrictions. An entity that has received a CAP message already 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 SIP Identity [RFC4474], 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 mandatory.
Note that none of the security mechanism in this document protect against a compromised sensor sending crafted alerts.

8. IANA Considerations

8.1. Registration of the 'application/cap+xml' MIME type

To:

ietf-types@iana.org

Subject: Registration of MIME media type application/ cap+xml

MIME media type name: application

MIME subtype name: cap+xml

Required parameters: (none)

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

Encoding considerations: Uses XML, which can employ 8-bit characters, depending on the character encoding used. See RFC 3023 [RFC3023], Section 3.2.

Security considerations: This content type is designed to carry payloads of the Common Alerting Protocol (CAP).

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.

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@nsn.com

Intended usage: Limited use

Author/Change controller:
IETF ECRIT working group

Other information: This media type is a specialization of application/xml RFC 3023 [RFC3023], and many of the considerations described there also apply to application/cap+xml.

8.2. 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.

8.3. IANA Registration of New AlertMsg-Error Header Field

Registry:

Header Name	compact	Reference
AlertMsg-Error		[this doc]

Header Field	Parameter Name	Predefined Values	Reference
AlertMsg-Error	code	yes	[this doc]

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 sip-parameters registry with two actions:

1. Update the Header Fields registry with

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

8.4. IANA Registration for the SIP AlertMsg-Error Codes

This document creates a new registry for SIP, called "AlertMsg-Error Codes". AlertMsg-Error codes provide reason for the error discovered by recipients, categorized by action to be taken by 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.

9. 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, and Marc Linsner for their review comments.

10. References

10.1. 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. 1.1 ", October 2005.
[RFC3023]	Murata, M., St. Laurent, S. and D. Kohn, "XML Media Types", RFC 3023, January 2001.
[RFC3629]	Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003.

[I-D.ietf-ecrit-phonebcp]	Rosen, B and J Polk, "Best Current Practice for Communications Services in support of Emergency Calling", Internet-Draft draft-ietf-ecrit-phonebcp-20, September 2011.
[I-D.ietf-sipcore-location-conveyance]	Polk, J, Rosen, B and J Peterson, "Location Conveyance for the Session Initiation Protocol", Internet-Draft draft-ietf-sipcore-location-conveyance-09, September 2011.
[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.
[RFC3311]	Rosenberg, J., "The Session Initiation Protocol (SIP) UPDATE Method", RFC 3311, October 2002.
[RFC3428]	Campbell, B., Rosenberg, J., Schulzrinne, H., Huitema, C. and D. Gurle, "Session Initiation Protocol (SIP) Extension for Instant Messaging", RFC 3428, December 2002.
[RFC6086]	Holmberg, C., Burger, E. and H. Kaplan, "Session Initiation Protocol (SIP) INFO Method and Package Framework", RFC 6086, January 2011.
[RFC3265]	Roach, A.B., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002.
[RFC3903]	Niemi, A., "Session Initiation Protocol (SIP) Extension for Event State Publication", RFC 3903, October 2004.
[RFC5234]	Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

10.2. Informative References

[I-D.ietf-atoca-requirements]	Schulzrinne, H, Norreys, S, Rosen, B and H Tschofenig, "Requirements, Terminology and Framework for Exigent Communications", Internet-Draft draft-ietf-atoca-requirements-02, October 2011.
[I-D.ietf-ecrit-trustworthy-location]	Tschofenig, H, Schulzrinne, H and B Aboba, "Trustworthy Location Information", Internet-Draft draft-ietf-ecrit-trustworthy-location-02, May 2011.
[RFC4474]	Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, August 2006.
[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.

Authors' Addresses

Brian Rosen Rosen NeuStar, Inc. 470 Conrad Dr Mars, PA 16046 US EMail: br@brianrosen.net

Henning Schulzrinne 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 Tschofenig Nokia Siemens Networks Linnoitustie 6 Espoo, 02600 Finland Phone: +358 (50) 4871445 EMail: Hannes.Tschofenig@gmx.net URI: http://www.tschofenig.priv.at