ATOCA	B. Rosen
Internet-Draft	NeuStar, Inc.
Intended status: Standards Track	H. Schulzrinne
Expires: May 28, 2011	Columbia U.
	H. Tschofenig
	Nokia Siemens Networks
	November 24, 2010

Session Initiation Protocol (SIP) Event Package for the Common Alerting Protocol (CAP)
draft-rosen-atoca-cap-01.txt

Abstract

The Common Alerting Protocol (CAP) is an XML document format for exchanging emergency alerts and public warnings. This document allows CAP documents to be distributed via the event notification mechanism available with the Session Initiation Protocol (SIP).

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

TOC

The Common Alerting Protocol (CAP) [cap] (Jones, E. and A. Botterell, “Common Alerting Protocol v. 1.1,” October 2005.) is an XML document format for exchanging emergency alerts and public warnings. The abstract architectural description for the distribution of alerts can be found in [I-D.ietf-atoca-requirements] (Schulzrinne, H., Norreys, S., Rosen, B., and H. Tschofenig, “Requirements, Terminology and Framework for Exigent Communications,” September 2010.).

This document specifies how CAP documents are distributed via the event notification mechanism available with the Session Initiation Protocol (SIP). Additionally, a MIME object is registered to allow CAP documents to be exchanged in other SIP messages.

2. Terminology

TOC

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] (Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels,” March 1997.).

Note that early warning specific definitions can be found in [I-D.ietf-atoca-requirements] (Schulzrinne, H., Norreys, S., Rosen, B., and H. Tschofenig, “Requirements, Terminology and Framework for Exigent Communications,” September 2010.).

3. The 'common-alerting-protocol' Event Package

TOC

RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.) defines a SIP extension for subscribing to remote nodes and receiving notifications of changes (events) in their states. It leaves the definition of many aspects of these events to concrete extensions, i.e. event packages. This document defines such a new "common-alerting-protocol" event package. RFC 3903 [RFC3903] (Niemi, A., “Session Initiation Protocol (SIP) Extension for Event State Publication,” October 2004.) defines an extension that allows SIP User Agents to publish event state. Event Publication Agents (EPA) use PUBLISH requests to inform an Event State Compositor (ESC) of changes in the "common-alerting-protocol event package. Acting as a notifier, the ESC notifies subscribers about emergency alerts and public warnings.

3.1. Package Name

TOC

The name of this package is "common-alerting-protocol". As specified in RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.), this value appears in the Event header field present in SUBSCRIBE and NOTIFY requests. As specified in RFC 3903 [RFC3903] (Niemi, A., “Session Initiation Protocol (SIP) Extension for Event State Publication,” October 2004.), this value also appears in the Event header field present in PUBLISH requests.

3.2. Event Package Parameters

TOC

RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.) allows event packages to define additional parameters carried in the Event header field. This event package does not define additional parameters.

3.3. SUBSCRIBE Bodies

TOC

RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.) allows a SUBSCRIBE request to contain a body. This document allows the body to contain event filters, see [RFC4660] (Khartabil, H., Leppanen, E., Lonnfors, M., and J. Costa-Requena, “Functional Description of Event Notification Filtering,” September 2006.) and [RFC4661] (Khartabil, H., Leppanen, E., Lonnfors, M., and J. Costa-Requena, “An Extensible Markup Language (XML)-Based Format for Event Notification Filtering,” September 2006.) with the information elements listed in the subsections below.

3.3.1. Location Filter

TOC

The 2D location shapes listed in [RFC5491] (Winterbottom, J., Thomson, M., and H. Tschofenig, “GEOPRIV Presence Information Data Format Location Object (PIDF-LO) Usage Clarification, Considerations, and Recommendations,” March 2009.) (e.g., <Point> <Polygon>, <Circle>, <Ellipse>, <ArcBand>) and the <civicAddress> element, defined in [RFC5139] (Thomson, M. and J. Winterbottom, “Revised Civic Location Format for Presence Information Data Format Location Object (PIDF-LO),” February 2008.). Repeating these elements is allowed and the semantic is equivalent to a union. The <alertArea> element indicates the area of interest; whenever an event happens in this area an alert message is delivered.

An example can be found below:

```
<?xml version="1.0" encoding="UTF-8"?>
<filter-set
    xmlns="urn:ietf:params:xml:ns:simple-filter"
    xmlns:af="urn:ietf:params:xml:ns:alert-filter"
    xmlns:gml="http://www.opengis.net/gml"
    xmlns:gs="http://www.opengis.net/pidflo/1.0">

    <filter id="123" uri="sip:presentity@example.com">
        <trigger>
            <af:alertArea>
                <gs:Circle
                    srsName="urn:ogc:def:crs:EPSG::4326">
                    <gml:pos>42.5463 -73.2512</gml:pos>
                    <gs:radius
                        uom="urn:ogc:def:uom:EPSG::9001">
                        5000
                    </gs:radius>
                </gs:Circle>
            </af:alertArea>
        </trigger>
    </filter>
</filter-set>
```

Example of a SIP SUBSCRIBE Body with a Location Filter

3.3.2. Service Filter

TOC

To filter different types of alerts the <serviceFilter> element MUST be included as a child element of the <what> element and it MUST list one or more Service URNs [RFC5031] (Schulzrinne, H., “A Uniform Resource Name (URN) for Emergency and Other Well-Known Services,” January 2008.), which indicate the type of alerts the recipient is interested in. This document registers a number of alerts relevant for exigent communications, which can be found in Section 6 (IANA Considerations).

An example can be found below:

```
<?xml version="1.0" encoding="UTF-8"?>
<filter-set
    xmlns="urn:ietf:params:xml:ns:simple-filter"
    xmlns:af="urn:ietf:params:xml:ns:alert-filter">
    <filter id="123" uri="sip:presentity@example.com">
        <what>
            <af:serviceFilter>
              urn:service:warning.met
            </af:serviceFilter>
        </what>
    </filter>
</filter-set>
```

Example of a SIP SUBSCRIBE Body with a Service Filter

3.3.3. Rate Control

TOC

[I-D.ietf-sipcore-event-rate-control] (Niemi, A., Kiss, K., and S. Loreto, “Session Initiation Protocol (SIP) Event Notification Extension for Notification Rate Control,” October 2010.) extends the SIP events framework by defining three "Event" header field parameters that allow a subscriber to set a minimum, a maximum and an average rate of event notifications generated by the notifier. This allows a subscriber to have overall control over the stream of notifications, for example to avoid being flooded.

A notifier is required to send a NOTIFY request immediately after creation of a subscription. If state is not available at that time, then the NOTIFY request may be sent with no content. A separate NOTIFY containing an alert message may be generated some time later when it becomes available. Figure 1 (SUBSCRIBE/NOTIFY with Rate Control) shows a SUBSCRIBE/NOTIFY exchange.

```
  Subscriber          Notifier
      |---SUBSCRIBE(1)--->|     Create subscription
      |<-------200--------|
      |<-----NOTIFY(2)----|     Return initial notify with no state
      |--------200------->|
      |<-----NOTIFY(3)----|     Alert message comes available
      |--------200------->|
```

Figure 1: SUBSCRIBE/NOTIFY with Rate Control

3.4. Subscription Duration

TOC

The default expiration time for subscriptions within this package is 3600 seconds. As per RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.), the subscriber MAY specify an alternate expiration in the Expires header field.

3.5. NOTIFY Bodies

TOC

As described in RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.), the NOTIFY message will contain bodies describing the state of the subscribed resource. This body is in a format listed in the Accept header field of the SUBSCRIBE request, or a package-specific default format if the Accept header field was omitted from the SUBSCRIBE request.

In this event package, the body of the notification contains a Common Alerting Protocol (CAP) document, i.e., an XML document. The format of the XML documents used by CAP are described in [cap] (Jones, E. and A. Botterell, “Common Alerting Protocol v. 1.1,” October 2005.).

For an initial notify, unlike for other event packages, there is no current initial state, unless there's a pending alert. Hence, returning a NOTIFY with a non-empty body only makes sense if there are indeed active alerts.

All subscribers and notifiers of the "common-alerting-protocol" event package MUST support the "application/common-alerting-protocol+xml" data format. The SUBSCRIBE request MAY contain an Accept header field. If no such header field is present, it has a default value of "application/common-alerting-protocol+xml" (assuming that the Event header field contains a value of "common-alerting-protocol"). If the Accept header field is present, it MUST include "application/common-alerting-protocol+xml".

3.6. Notifier Processing of SUBSCRIBE Requests

TOC

The contents of a CAP document may contain public information, depending on the alert message type and the intended recipient of the alert message. It is, however, expected that in many cases providing CAP documents does not require authorization by subscribers.

3.7. Notifier Generation of NOTIFY Requests

TOC

RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.) details the formatting and structure of NOTIFY messages. However, packages are mandated to provide detailed information on when to send a NOTIFY, how to compute the state of the resource, how to generate neutral or fake state information, and whether state information is complete or partial. This section describes those details for the common-alerting-protocol event package.
A notifier MAY send a NOTIFY at any time. Typically, it will send one when an alert or early warning message is available. The NOTIFY request contains a body containing one or multiple CAP document(s). The times at which the NOTIFY is sent for a particular subscriber, and the contents of the body within that notification, are subject to any rules specified by the authorization policy that governs the subscription.
If the subscription is rejected, a NOTIFY MAY be sent. As described in RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.), the Subscription-State header field indicates the state of the subscription.
The body of the NOTIFY MUST be sent using one of the types listed in the Accept header field in the most recent SUBSCRIBE request, or using the type "application/common-alerting-protocol+xml" if no Accept header field was present.
Notifiers act as Event State Compositors (ESC). Thus, they learn the 'common-alerting-protocol' event state via PUBLISH requests sent from authorized Event Publication Agents (EPAs). A Notifier may also be an EPA, or might accept PUBLISH requests from authorized EPAs.

TOC

3.8. Subscriber Processing of NOTIFY Requests

RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.) leaves it to event packages to describe the process followed by the subscriber upon receipt of a NOTIFY request, including any logic required to form a coherent resource state.

TOC

3.9. Handling of Forked Requests

RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.) requires each package to describe handling of forked SUBSCRIBE requests.
Given that a single SUBSCRIBE might include multiple services and locations, it is reasonable and useful for a SUBSCRIBE request to fork and to reach multiple UAs. This is equivalent to multiple sources providing alerts for the same geographical, with a dedicated relay serving an aggregation function.
As a result, a forked SUBSCRIBE request can install multiple subscriptions. Subscribers to this package MUST be prepared to install subscription state for each NOTIFY generated as a result of a single SUBSCRIBE.

TOC

3.10. Rate of Notifications

RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.) requires each package to specify the maximum rate at which notifications can be sent.
Notifiers SHOULD NOT generate notifications for a single user at a rate of more than once every five seconds.

TOC

3.11. State Agents

RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.) requires each package to consider the role of state agents in the package and, if they are used, to specify how authentication and authorization are done. This specification allows state agents to be located in the network.

TOC

3.12. Examples

An example is provided in Section 4 (Examples).

3.13. Use of URIs to Retrieve State

TOC

RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.) allows packages to use URIs to retrieve large state documents.
CAP documents are fairly small. This event package does not provide a mechanism to use URIs to retrieve large state documents.

3.14. PUBLISH Bodies

TOC

RFC 3903 [RFC3903] (Niemi, A., “Session Initiation Protocol (SIP) Extension for Event State Publication,” October 2004.) requires event packages to define the content types expected in PUBLISH requests.
In this event package, the body of a PUBLISH request may contain a CAP document. A CAP document describes an emergency alert or an early warning event.
All EPAs and ESCs MUST support the "application/common-alerting-protocol+xml" data format and MAY support other formats.

3.15. PUBLISH Response Bodies

TOC

This specification assumes that the response to a PUBLISH does not contain a body.

3.16. Multiple Sources for Event State

TOC

RFC 3903 [RFC3903] (Niemi, A., “Session Initiation Protocol (SIP) Extension for Event State Publication,” October 2004.) requires event packages to specify whether multiple sources can contribute to the event state view at the ESC.
This event package allows different EPAs to publish CAP documents for a particular user. The concept of composition is not applicable for this application usage.

3.17. Event State Segmentation

TOC

RFC 3903 [RFC3903] (Niemi, A., “Session Initiation Protocol (SIP) Extension for Event State Publication,” October 2004.) defines segments within a state document. Each segment is defined as one of potentially many identifiable sections in the published event state.
This event package defines does not differentiate between different segments.

3.18. Rate of Publication

TOC

RFC 3903 [RFC3903] (Niemi, A., “Session Initiation Protocol (SIP) Extension for Event State Publication,” October 2004.) allows event packages to define their own rate of publication. This event package allows rate control to be utilized, as described in Section 3.3.3 (Rate Control).

4. Examples

TOC

Here is an example of a CAP document.

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

<alert xmlns="urn:oasis:names:tc:emergency:cap:1.1">
    <identifier>KSTO1055887203</identifier>
    <sender>KSTO@NWS.NOAA.GOV</sender>
    <sent>2003-06-17T14:57:00-07:00</sent>
    <status>Actual</status>
    <msgType>Alert</msgType>
    <scope>Public</scope>
    <info>
        <category>Met</category>
        <event>SEVERE THUNDERSTORM</event>
        <urgency>Severe</urgency>
        <certainty>Likely</certainty>
        <senderName>NATIONAL WEATHER SERVICE SACRAMENTO</senderName>
        <headline>SEVERE THUNDERSTORM WARNING</headline>
        <description> AT 254 PM PDT...
            NATIONAL WEATHER SERVICE
            DOPPLER RADAR INDICATED A SEVERE
            THUNDERSTORM OVER SOUTH CENTRAL ALPINE COUNTY...
            OR ABOUT 18 MILES SOUTHEAST OF
            KIRKWOOD... MOVING SOUTHWEST AT 5 MPH. HAIL...
            INTENSE RAIN AND STRONG DAMAGING WINDS
            ARE LIKELY WITH THIS STORM </description>
        <instruction> TAKE COVER IN A SUBSTANTIAL SHELTER
            UNTIL THE STORM PASSES </instruction>
        <contact>BARUFFALDI/JUSKIE</contact>
        <area>
            <areaDesc> EXTREME NORTH CENTRAL TUOLUMNE COUNTY
                IN CALIFORNIA, EXTREME NORTHEASTERN
                CALAVERAS COUNTY IN CALIFORNIA, SOUTHWESTERN
                ALPINE COUNTY IN CALIFORNIA </areaDesc>
            <polygon> 38.47,-120.14 38.34,-119.95 38.52,-119.74
                38.62,-119.89 38.47,-120.14 </polygon>
        </area>
    </info>
</alert>
```

Example for a Severe Thunderstorm Warning

TOC

5. Security Considerations

This section discusses security considerations when using SIP to distribute warning messages using CAP.
Based on the framework outlined in [I-D.ietf-atoca-requirements] (Schulzrinne, H., Norreys, S., Rosen, B., and H. Tschofenig, “Requirements, Terminology and Framework for Exigent Communications,” September 2010.) the following security concerns arise:

> **Amplification Attacks:** An adversary could inject alerts into the message handling system and therefore a single PUBLISH request could potentially results in millions of NOTIFY messages delivered to receivers. Injecting messages may happen at a number of ways, such as by adversaries who manage to impersonate a legitimate originator, a relay or gateway. Ensuring that only authorized entities are permitted to inject alerts is a pre-condition. This does, however, not help if the host of a trusted participant in the message handling system got compromised.
>
> **Forgery of Alerts:** Alerts may get modified or replayed. The former is possible if the adversary manages to get access to a relay or gateway. Two mechanisms are proposed for countering forgery: using digital signatures or channel security (TLS). The first provides end-to-end security; the second utilizes a hop by hop security model based on a transitive chain of trust.

The sub-sections below discuss these threats and their countermeasures in more detail.

5.1. Amplification

TOC

> **Threat:**
> The attacker could then conceivably attempt to impersonate an originator, or a relay. A side effect of being able to inject an alert for distribution is the amplification effect.
>
> **Countermeasures:**
> When an entity receives a CAP message it has to determine whether the entity distributing the CAP messages is genuine to avoid accepting messages that are injected by malicious entities.
>
> When receiving a CAP document a couple of verification steps must be performed. First, it needs to be ensured that the message was delivered via a trusted entity and that the communication channel between the User Agent and it's SIP proxy is properly secured to

exclude various attacks at the SIP level. Then, the message contains the <sender> that may contain an entity that falls within the white list of the entity receiving the message. Finally, the message is protected by a digital signature and the entity signing the CAP message may again be listed in a white list of the receiving entity and may therefore be trusted. If none of these verification checks lead to a positive indication of a known sender then the CAP document should be treated as suspicious and configuration at the receiving entity may dictate how to process and display CAP documents in such a case.

5.1.1. Forgery of Alerts

TOC

Threat:
A malicious user could forge a CAP document. Alternatively, a CAP document distributed earlier could be replied.

Countermeasures:
To avoid forgery, the entities must assure that proper mechanisms for protecting the CAP documents are employed, for example signing the CAP document itself or securing the communication between participating entities using TLS. Section 3.3.2.1 of [cap] (Jones, E. and A. Botterell, “Common Alerting Protocol v. 1.1,” October 2005.) specifies the signing of CAP documents.

Regarding replay attacks the following observations can be made. 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 originator/author 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. Recipients who enter the area of a disaster after the initial distribution of warnings may not yet have seen the original CAP message and, as such, would not be able to distinguish a replay from the initial message being sent around. However, if the threat that lead to the distribution of warning messages is still imminent then there is no reason not to worry about that message. The originator/author distributing the alert is, however, adviced to carefully select a value for the <expires> element and it is RECOMMENDED to set a value for this element.

6. IANA Considerations

TOC

6.1. Registration of the 'common-alerting-protocol' Event Package

TOC

This specification registers an event package, based on the registration procedures defined in RFC 3265 [RFC3265] (Roach, A., “Session Initiation Protocol (SIP)-Specific Event Notification,” June 2002.). The following is the information required for such a registration:

> **Package Name:** common-alerting-protocol
>
> **Package or Template-Package:** This is a package.
>
> **Published Document:** RFC XXX [Replace by the RFC number of this specification].
>
> **Person to Contact:** Hannes Tschofenig, Hannes.Tschofenig@nsn.com

6.2. Registration of the 'application/common-alerting-protocol+xml' MIME type

TOC

To:
ietf-types@iana.org

Subject: Registration of MIME media type application/ common-alerting-protocol+xml

MIME media type name: application

MIME subtype name: common-alerting-protocol+xml

Required parameters: (none)

Optional parameters: charset; Indicates the character encoding of enclosed XML. Default is UTF-8 [RFC3629] (Yergeau, F., “UTF-8, a transformation format of ISO 10646,” November 2003.).

Encoding considerations: Uses XML, which can employ 8-bit characters, depending on the character encoding used. See RFC 3023 [RFC3023] (Murata, M., St. Laurent, S., and D. Kohn, “XML Media Types,” January 2001.), 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 early warnings according to the CAP standard.

Additional information: OASIS has published the Common Alerting Protocol at [cap] (Jones, E. and A. Botterell, “Common Alerting Protocol v. 1.1,” October 2005.).

Person & email address to contact for further information: Hannes Tschofenig, Hannes.Tschofenig@nsn.com

Intended usage: Limited use

Author/Change controller: IETF SIPPING working group

Other information: This media type is a specialization of application/xml RFC 3023 [RFC3023] (Murata, M., St. Laurent, S., and D. Kohn, “XML Media Types,” January 2001.), and many of the considerations described there also apply to application/common-alerting-protocol+xml.

6.3. Early Warning Service URNs

TOC

In according with RFC 5031 this document defines a new top-level service called 'warning'. This section defines the first service registration within the IANA registry using the top-level service label 'warning'.

The 'warning' service type describes emergency services requiring an immediate action or remedy by the recipient of the alert message as instructed by the author of the message. Additional sub-services can be added after expert review and must be of general public interest and have a similar emergency nature. The expert is designated by the ECRIT working group, its successor, or, in their absence, the IESG. The expert review should only approve emergency services that are offered widely and in different countries, with approximately the same caller expectation in terms of services rendered.

The following list contains the initial IANA registration for the 'warning' service.

> **urn:service:warning.geo:** Geophysical (inc. landslide)
>
> **urn:service:warning.met:** Meteorological (inc. flood)
>
> **urn:service:warning.safety:** General emergency and public safety
>
> **urn:service:warning.security:** Law enforcement, military, homeland and local/private security
>
> **urn:service:warning.rescue:** Rescue and recovery
>
> **urn:service:warning.fire:** Fire suppression and rescue
>
> **urn:service:warning.health:** Medical and public health
>
> **urn:service:warning.env:** Pollution and other environmental
>
> **urn:service:warning.transport:** Public and private transportation
>
> **urn:service:warning.infra:** Utility, telecommunication, other non-transport infrastructure
>
> **urn:service:warning.cbrne:** Chemical, Biological, Radiological, Nuclear or High-Yield Explosive threat or attack
>
> **urn:service:warning.other:** Other events

7. Open Issues

TOC

The authors would like to point to a number of issues that require discussion:

Rate Control: The -00 version of the document introduced rate control for notifications Section 3.3.3 (Rate Control). Is this functionality is needed?

Early Warning Service URNs: Specifying services is always difficult since there is no universally agreed service semantic. This document contains a proposal that re-use the classification in the CAP specification [cap] (Jones, E. and A. Botterell, “Common Alerting Protocol v. 1.1,” October 2005.). Is the proposal acceptable?

Event Filter: By using [RFC4660] (Khartabil, H., Leppanen, E., Lonnfors, M., and J. Costa-Requena, “Functional Description of Event Notification Filtering,” September 2006.) and [RFC4661] (Khartabil, H., Leppanen, E., Lonnfors, M., and J. Costa-Requena, “An Extensible Markup Language (XML)-Based Format for Event Notification Filtering,” September 2006.) filters in the body of a SUBSCRIBE the number of notifications can be reduced to those of interest to the subscriber. There is a certain overhead associated with the generic usage of those event filters. Should alternatives be considered?

Forked SUBSCRIBE Requests: This document allows forked subscribe request. This is useful when a single service is offered by more than one entity and therefore related to the cases discussed in [I-D.forte-lost-extensions] (Forte, A. and H. Schulzrinne, “Location-to-Service Translation Protocol (LoST) Extensions,” September 2010.) and in [I-D.forte-ecrit-service-classification] (Forte, A. and H. Schulzrinne, “Labels for Common Location-Based Services,” January 2010.). For example, imagine a warning service like 'urn:service:warning.geo' that is advertised by a number of different service providers.

Security: The security consideration section was re-written and focuses now mostly on two types of attacks, namely amplificiation and forgery. Does this reflect the understanding of the group?

TOC

8. Acknowledgments

The authors would like to thank Cullen Jennings for his early support and the participants of the Early Warning Adhoc meeting at IETF#69 for their feedback.
We would furthermore like to thank Martin Thomson for his detailed draft review in July 2010.

9. References

TOC

9.1. Normative References

TOC

[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.
[RFC3265]	Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification," RFC 3265, June 2002 (TXT).
[RFC3903]	Niemi, A., "Session Initiation Protocol (SIP) Extension for Event State Publication," RFC 3903, October 2004 (TXT).
[RFC3023]	Murata, M., St. Laurent, S., and D. Kohn, "XML Media Types," RFC 3023, January 2001 (TXT).
[RFC3629]	Yergeau, F., "UTF-8, a transformation format of ISO 10646," STD 63, RFC 3629, November 2003 (TXT).
[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 (TXT).
[RFC5139]	Thomson, M. and J. Winterbottom, "Revised Civic Location Format for Presence Information Data Format Location Object (PIDF-LO)," RFC 5139, February 2008 (TXT).
[RFC5031]	Schulzrinne, H., "A Uniform Resource Name (URN) for Emergency and Other Well-Known Services," RFC 5031, January 2008 (TXT).
[I-D.ietf-sipcore-event-rate-control]	Niemi, A., Kiss, K., and S. Loreto, "Session Initiation Protocol (SIP) Event Notification Extension for Notification Rate Control," draft-ietf-sipcore-event-rate-control-05 (work in progress), October 2010 (TXT).

[RFC4660]	Khartabil, H., Leppanen, E., Lonnfors, M., and J. Costa-Requena, “Functional Description of Event Notification Filtering,” RFC 4660, September 2006 (TXT).
[RFC4661]	Khartabil, H., Leppanen, E., Lonnfors, M., and J. Costa-Requena, “An Extensible Markup Language (XML)-Based Format for Event Notification Filtering,” RFC 4661, September 2006 (TXT).

9.2. Informative References

TOC

[I-D.ietf-atoca-requirements]	Schulzrinne, H., Norreys, S., Rosen, B., and H. Tschofenig, “Requirements, Terminology and Framework for Exigent Communications,” draft-ietf-atoca-requirements-00 (work in progress), September 2010 (TXT).
[I-D.forte-lost-extensions]	Forte, A. and H. Schulzrinne, “Location-to-Service Translation Protocol (LoST) Extensions,” draft-forte-lost-extensions-02 (work in progress), September 2010 (TXT).
[I-D.forte-ecrit-service-classification]	Forte, A. and H. Schulzrinne, “Labels for Common Location-Based Services,” draft-forte-ecrit-service-classification-03 (work in progress), January 2010 (TXT).

Authors' Addresses

TOC

Brian Rosen
NeuStar, Inc.
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
Nokia Siemens Networks
Linnoitustie 6
Espoo 02600
Finland
Phone: +358 (50) 4871445
Email: Hannes.Tschofenig@gmx.net
URI: http://www.tschofenig.priv.at