

Internet Engineering Task Force (IETF)
Request for Comments: 6447
Category: Standards Track
ISSN: 2070-1721

R. Mahy
Individual
B. Rosen
NeuStar
H. Tschofenig
Nokia Siemens Networks
January 2012

Filtering Location Notifications in the Session Initiation Protocol (SIP)

Abstract

This document describes filters that limit asynchronous location notifications to compelling events. These filters are designed as an extension to RFC 4661, an XML-based format for event notification filtering, and based on RFC 3856, the SIP presence event package. The resulting location information is conveyed in existing location formats wrapped in the Presence Information Data Format Location Object (PIDF-LO).

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <http://www.rfc-editor.org/info/rfc6447>.

Copyright Notice

Copyright (c) 2012 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.	Filter Definitions	4
3.1.	Movement	4
3.2.	Speed Changes	5
3.3.	Element Value Changes	5
3.4.	Entering or Exiting a Region	8
3.5.	Location Type	10
3.6.	Rate Control	12
4.	XML Schema	14
5.	Security Considerations	15
6.	IANA Considerations	16
6.1.	URN Sub-Namespace Registration for urn:ietf:params:xml:ns:location-filter	16
6.2.	Schema Registration for location-filter	16
7.	Contributors	17
8.	Acknowledgments	17
9.	References	17
9.1.	Normative References	17
9.2.	Informative References	18

1. Introduction

Conveying location information encapsulated with a Presence Information Data Format Location Object (PIDF-LO) [RFC4119] document within SIP is described in [SIP-LOC]. An alternative signaling approach to location conveyance, which uses asynchronous communication, is available with the SIP event notification mechanisms (see RFC 3265 [RFC3265]). This approach conveys location information in PIDF-LO format using the presence event package [RFC3856]. This document focuses on the event notification paradigm.

Determining when to send event notifications with location information is technically more challenging than deciding when to send other categories of notifications, since location may be measured as a continuous gradient. Unlike notifications using discrete-valued quantities, it is difficult to know when a change in location is sufficiently large to warrant a notification. Event notifications [RFC3265] can be used with filters (see RFC 4661 [RFC4661]) that allow the number of notifications to be reduced. The mechanism described in this document defines an extension to RFC 4661 [RFC4661], which limits location notification to events that are of relevance to the subscriber. These filters persist until they are replaced with a newer filter or until the subscription itself is terminated.

The frequency of notifications necessary for various geographic location applications varies dramatically. The subscriber should be able to get asynchronous notifications with appropriate frequency and granularity, without being flooded with a large number of notifications that are not important to the application.

This document defines new event filters and describes others using existing mechanisms that may be relevant to a subscriber in the context of location filtering. Based on the functionality defined in this document, notifications can be provided in the following cases:

1. the Target moves more than a specified distance since the last notification (see Section 3.1).
2. the Target exceeds a specified speed (see Section 3.2).
3. the Target enters or exits a 2-dimensional region, described by a circle or a polygon (see Section 3.4).
4. one or more of the values of the specified civic location have changed for the location of the Target (see Section 3.3). For example, the value of the civic address '<A1>' element has changed from 'California' to 'Nevada'.

5. the type of location information requested (see Section 3.5) changes, for example, from civic to geodetic location or vice versa.
6. a certain amount of time passes (see Section 3.6).

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 reuses terminology from [RFC6280].

3. Filter Definitions

This specification builds on a number of other specifications, as noted in Section 1. In order to reduce the number of options (and thereby decrease the chance of interoperability problems), the functionality described in the following sub-sections of [RFC4661] MUST be implemented: the <ns-bindings> element (see Section 3.3 of [RFC4661]); the <filter> element (Section 3.4 of [RFC4661]); and the <trigger> element (Section 3.6 of [RFC4661]), except for the <added> and <removed> sub-elements.

3.1. Movement

The <moved> element MUST contain a value in meters indicating the minimum distance that the resource must have moved from the location of the resource since the last notification was sent in order to trigger this event. The distance MUST be measured in meters absolutely from the point of the last notification, and must include vertical movement. The <moved> element MUST NOT appear more than once as a child element of the <filter> element.

```
<?xml version="1.0" encoding="UTF-8"?>
<filter-set
  xmlns="urn:ietf:params:xml:ns:simple-filter"
  xmlns:lf="urn:ietf:params:xml:ns:location-filter">
  <filter id="123" uri="sip:presentity@example.com">
    <trigger>
      <lf:moved>300</lf:moved>
    </trigger>
  </filter>
</filter-set>
```

Figure 1: Movement Filter Example

3.2. Speed Changes

Speed changes can be filtered by combining functionality from RFC 4661 with the PIDF-L0 extensions for spatial orientation, speed, heading, and acceleration defined in [RFC5962]. The value of the `<speed>` element from [RFC5962] MUST be defined in meters per second. Note that the condition could be met by a change in any axis, including altitude.

Figure 2 shows an example for a trigger that fires when the speed of the Target changes by 3 meters per second.

```
<?xml version="1.0" encoding="UTF-8"?>
<filter-set xmlns="urn:ietf:params:xml:ns:simple-filter">
  <ns-bindings>
    <ns-binding prefix="dyn"
      urn="urn:ietf:params:xml:schema:pidf:dynamic"/>
  </ns-bindings>
  <filter id="123" uri="sip:presentity@example.com">
    <trigger>
      <changed by="3">
        //dyn:speed
      </changed>
    </trigger>
  </filter>
</filter-set>
```

Figure 2: Speed Change Example

An implementation MUST support `<ns-bindings>` to replace the namespace prefix. The XPath expression MUST start with a `'//'` followed by a single element. No other form of XPath expression is supported. The `<changed>` element comes with a few attributes but only the `'by'` attribute MUST be implemented by this specification.

3.3. Element Value Changes

Changes in values, for example related to civic location information, is provided by the base functionality offered with RFC 4661 utilizing the `<changed>` element.

The following example illustrates a filter that triggers when the Target's location changes from 'FR' (France) to some other country.

```
<?xml version="1.0" encoding="UTF-8"?>
<filter-set xmlns="urn:ietf:params:xml:ns:simple-filter">
  <ns-bindings>
    <ns-binding prefix="ca"
      urn="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"/>
  </ns-bindings>
  <filter id="123" uri="sip:presentity@example.com">
    <trigger>
      <changed from="FR">//ca:country</changed>
    </trigger>
  </filter>
</filter-set>
```

Figure 3: Element Value Change Example (Country Change)

At times when it is desirable to know if any one element of a list of CAtypes changes, then they have to be put into separate <changes> filters to ensure the subscriber is notified when any of the element values change. Figure 4 shows such an example that illustrates the difference.

(A change in value of ANY of the five tokens triggers an event.)

```
<?xml version="1.0" encoding="UTF-8"?>
<filter-set xmlns="urn:ietf:params:xml:ns:simple-filter">
  <ns-bindings>
    <ns-binding prefix="ca"
      urn="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"/>
  </ns-bindings>
  <filter id="123" uri="sip:presentity@example.com">
    <trigger>
      <changed>//ca:country</changed>
    </trigger>
    <trigger>
      <changed>//ca:A1</changed>
    </trigger>
    <trigger>
      <changed>//ca:A2</changed>
    </trigger>
    <trigger>
      <changed>//ca:A3</changed>
    </trigger>
    <trigger>
      <changed>//ca:PC</changed>
    </trigger>
  </filter>
</filter-set>
```

Figure 4: Element Value Change Example

Finally, Figure 5 shows an example where a notification is sent when the civic address tokens A3 and PC change (BOTH elements must change in order to let the <trigger> element evaluate to TRUE).

(Only a change in BOTH tokens triggers an event.)

```
<?xml version="1.0" encoding="UTF-8"?>
<filter-set xmlns="urn:ietf:params:xml:ns:simple-filter">
  <ns-bindings>
    <ns-binding prefix="ca"
      urn="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"/>
  </ns-bindings>
  <filter id="123" uri="sip:presentity@example.com">
    <trigger>
      <changed>//ca:A3</changed>
      <changed>//ca:PC</changed>
    </trigger>
  </filter>
</filter-set>
```

Figure 5: Element Value Change Example

Note: The civic address tokens country, A1, A2, ..., A6 are hierarchical. It is likely that a change in one civic address token therefore leads to changes of tokens lower in the hierarchy, e.g., a change in A3 ('city or town') may cause a change in A4, A5, and A6.

An implementation **MUST** support `<ns-bindings>` to replace the namespace prefix. The XPath expression **MUST** start with a `'//'` followed by a single element. No other form of XPath expression is supported. No other variant is supported. The `<changed>` element comes with a few attributes and the `'by'`, `'to'`, and `'from'` attribute **MUST** be implemented to support this specification.

3.4. Entering or Exiting a Region

The `<enterOrExit>` condition is satisfied when the Target enters or exits a 2-dimensional region described by a polygon (as defined in Section 5.2.2 of [RFC5491]) or a circle (as defined in Section 5.2.3 of [RFC5491]). The `<enterOrExit>` element **MUST** contain either a polygon or a circle as a child element. The `<enterOrExit>` element **MUST NOT** have more than one polygon and/or circle.

If the Target was previously outside the region, the notifier sends a notification when the Target's location is within the region with at least 50% confidence. Similarly, when a Target starts within the region, a notification is sent when the Target's location moves outside the region with at least 50% confidence.

Note that having 50% confidence that the Target is inside the area does not correspond to 50% outside. The confidence that the location is within the region, plus the confidence that the location is

outside the region is limited to the confidence of the location. The total confidence depends on the confidence in the location, which is always less than 100% (95% is recommended in [RFC5491]). The benefit of this is that notifications are naturally limited: small movements (relative to the uncertainty of the location) at the borders of the region do not trigger notifications.

Figure 6 shows filter examples whereby a notification is sent when the Target enters or exits an area described by a circle, and Figure 7 describes an area using a polygon.

```
<?xml version="1.0" encoding="UTF-8"?>
<filter-set
  xmlns="urn:ietf:params:xml:ns:simple-filter"
  xmlns:lf="urn:ietf:params:xml:ns:location-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>
      <lf:enterOrExit>
        <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">
                850.24
            </gs:radius>
          </gs:Circle>
        </lf:enterOrExit>
      </trigger>
    </filter>
  </filter-set>
```

Figure 6: <enterOrExit> Circle Filter Example

```

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

  <filter id="123" uri="sip:presentity@example.com">
    <trigger>
      <lf:enterOrExit>
        <gml:Polygon srsName="urn:ogc:def:crs:EPSG::4326">
          <gml:exterior>
            <gml:LinearRing>
              <gml:pos>43.311 -73.422</gml:pos>
              <!--A-->
              <gml:pos>43.111 -73.322</gml:pos>
              <!--F-->
              <gml:pos>43.111 -73.222</gml:pos>
              <!--E-->
              <gml:pos>43.311 -73.122</gml:pos>
              <!--D-->
              <gml:pos>43.411 -73.222</gml:pos>
              <!--C-->
              <gml:pos>43.411 -73.322</gml:pos>
              <!--B-->
              <gml:pos>43.311 -73.422</gml:pos>
              <!--A-->
            </gml:LinearRing>
          </gml:exterior>
        </gml:Polygon>
      </lf:enterOrExit>
    </trigger>
  </filter>
</filter-set>

```

Figure 7: <enterOrExit> Polygon Filter Example

3.5. Location Type

The <locationType> element MAY be included as a child element of the <what> element. It contains a list of location information types that are requested by the subscriber. The following list describes the possible values:

any: The Notifier SHOULD attempt to provide location information in all forms available to it.

geodetic: The Notifier **SHOULD** return a location by value in the form of a geodetic location.

civic: The Notifier **SHOULD** return a location by value in the form of a civic address.

The Notifier **SHOULD** return the requested location type or types. The location types the Notifier returns also depends on the setting of the optional 'exact' attribute. If the 'exact' attribute is set to "true", then the Notifier **MUST** return either the requested location type or no location information. The 'exact' attribute does not apply (is ignored) for a request for a location type of "any".

In the case of a request for specific locationType(s) and the 'exact' attribute is "false", the Notifier **MAY** provide additional location types, or it **MAY** provide alternative types if the request cannot be satisfied for a requested location type.

If the <locationType> element is absent, a value of "any" **MUST** be assumed as the default.

The Notifier **SHOULD** provide civic and geodetic location information in the response in the same order in which they were included in the "locationType" element in the request, if both were explicitly requested. Indeed, the primary advantage of including specific location types in a request when the 'exact' attribute is set to "false" is to ensure that one receives the available locations in a specific order. For example, a subscription for "civic" (with the 'exact' attribute set to "false") could yield any of the following location types in the response:

- o civic
- o civic, geodetic
- o geodetic (only if civic is not available)

The default value of "false" for the 'exact' attribute allows the Notifier the option of returning something beyond what is specified, such as a set of location URIs when only a civic location was requested.

An example is shown in Figure 8 that utilizes the <locationType> element with the 'exact' attribute.

```
<?xml version="1.0" encoding="UTF-8"?>
<filter-set
  xmlns="urn:ietf:params:xml:ns:simple-filter"
  xmlns:lf="urn:ietf:params:xml:ns:location-filter">
  <filter id="123" uri="sip:presentity@example.com">
    <what>
      <lf:locationType exact="true">
        geodetic
      </lf:locationType>
    </what>
  </filter>
</filter-set>
```

Figure 8: <locationType> Filter Example

3.6. Rate Control

[RFC6446] extends the SIP events framework by defining three Event header field parameters that allow a subscriber to set a minimum, a maximum, and an adaptive minimum 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. Two of the parameters, namely "min-rate" (which specifies a minimum notification rate per second) and "max-rate" (which specifies a maximum notification rate per second) are used by this document. Only the implementation of these two attributes is required from the attributes defined in [RFC6446]. Whenever the time since the most recent notification exceeds the interval corresponding to $1 / \text{"min-rate"}$, the current state would be sent in its entirety, just like after a subscription refresh.

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 location is subsequently generated so that the rate of notification since the last NOTIFY falls between "min-rate" and "max-rate". An important use case for location-based applications focuses on the behavior of the initial NOTIFY message(s) and the information it returns, for example in case of emergency call routing. When an initial NOTIFY is transmitted, it might not include complete state.

Subscriber	Notifier
---SUBSCRIBE(1)--->	Create subscription (w/large value
<-----200-----	for min-rate and max-rate)
<-----NOTIFY(2)----	Return initial notify with no state
-----200----->	
<-----NOTIFY(3)----	Return full state (between min-rate
-----200----->	and max-rate)
---SUBSCRIBE(4)--->	Update subscription (to update
<-----200-----	min-rate and max-rate)

Figure 9: SUBSCRIBE/NOTIFY with Rate Control

Figure 9 shows a SUBSCRIBE/NOTIFY exchange. The initial SUBSCRIBE message (1) has filters attached and contains a "min-rate" rate control parameter. In certain situations, it is important to obtain some amount of location information within a relatively short and pre-defined period of time, even if the obtained location information contains a high amount of uncertainty and location information with less uncertainty could be available at a later point in time. An example is emergency call routing where an emergency services routing proxy may need to obtain location information suitable for routing rather quickly and subsequently a Public Safety Answering Point requests location information for dispatch.

To obtain location information in a timely fashion using the SUBSCRIBE/NOTIFY mechanism, it is RECOMMENDED that the initial SUBSCRIBE contain a "min-rate" rate control parameter (with a large value, corresponding to a very short delay before the next notification) that is updated in a later message to a more sensible value. This provides equivalent functionality to the 'responseTime' attribute in Section 6.1 of [RFC5985]. The "min-rate" for this first request is therefore much larger (much more rapid) than the updated "min-rate" value. Depending on the value in the "min-rate" parameter, the Notifier may immediately send the initial NOTIFY message (see message 2) without a body if no location information is available at this point in time. The desired location information may then arrive in the subsequent NOTIFY message (see message 3). Updating the "min-rate" for the subscription can be performed in the 200 response (see message 3) to the NOTIFY that contains location state, or in a subsequent SUBSCRIBE request (as in message 4).

4. XML Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="urn:ietf:params:xml:ns:location-filter"
  xmlns:filter="urn:ietf:params:xml:ns:location-filter"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:gml="http://www.opengis.net/gml">

  <xs:element name="enterOrExit" type="gml:GeometryPropertyType"/>

  <xs:element name="moved" type="filter:movedType"/>

  <xs:complexType name="movedType">
    <xs:simpleContent>
      <xs:extension base="xs:double">
        <xs:anyAttribute namespace="##any" processContents="lax"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>

  <xs:element name="locationType" type="filter:locationTypeType"/>

  <xs:simpleType name="locationTypeBase">
    <xs:union>
      <xs:simpleType>
        <xs:restriction base="xs:token">
          <xs:enumeration value="any"/>
        </xs:restriction>
      </xs:simpleType>
      <xs:simpleType>
        <xs:restriction base="filter:locationTypeList">
          <xs:minLength value="1"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:union>
  </xs:simpleType>

  <xs:simpleType name="locationTypeList">
    <xs:list>
      <xs:simpleType>
        <xs:restriction base="xs:token">
          <xs:enumeration value="civic"/>
          <xs:enumeration value="geodetic"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:list>
  </xs:simpleType>

</xs:schema>
```

```
<xs:complexType name="locationTypeType">
  <xs:simpleContent>
    <xs:extension base="filter:locationTypeBase">
      <xs:attribute name="exact" type="xs:boolean"
        use="optional" default="false"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
</xs:schema>
```

Figure 10: XML Schema

5. Security Considerations

This document specifies one element, namely filters, utilized in larger systems. As such, this document builds on a number of specifications for the security of the complete solution, namely

- o the SIP event notification mechanism, described in RFC 3265 [RFC3265], defining the SUBSCRIBE/NOTIFY messages.
- o the presence event package, described in RFC 3856 [RFC3856], which is a concrete instantiation of the general event notification framework.
- o the filter framework, described in RFC 4661 [RFC4661], to offer the ability to reduce the amount of notifications being sent.

Finally, this document indirectly (via the SIP presence event package) relies on PIDF-LO, described in RFC 4119 [RFC4119], as the XML container that carries location information.

Each of the documents listed above comes with a Security Considerations section but the security and privacy aspects are best covered by the SIP presence event package; see Section 9 of [RFC3856], and with the GEOPRIV architectural description found in [RFC6280].

The functionality offered by authorization policies to limit access to location information is provided by other protocols, such as Common Policy [RFC4745], Geolocation Policy [GEO-POLICY], or more recent work around HTTP-Enabled Location Delivery (HELD) context [HELD]. Although [GEO-POLICY] defines a standardized format for geolocation authorization policies, it does not define specific policies for controlling filters.

The functionality described in this document extends the filter framework with location-specific filters. Local policies might be

associated with the usage of certain filter constructs and with the amount of notifications that specific filter settings might cause. Uploading filters have a significant effect on the ways in which the request is handled at a server. As a result, it is especially important that messages containing this extension be authenticated and authorized. RFC 4661 [RFC4661] discusses this security threat and proposes authentication and authorization solutions applicable to this specification.

6. IANA Considerations

6.1. URN Sub-Namespace Registration for urn:ietf:params:xml:ns:location-filter

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

URI: urn:ietf:params:xml:ns:location-filter

Registrant Contact: IETF, GEOPRIV working group, <geopriv@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>Location Filter Namespace</title>
</head>
<body>
  <h1>Namespace for PIDF-L0 Location Filters</h1>
  <h2>urn:ietf:params:xml:ns:location-filter</h2>
  <p>See <a href="http://www.rfc-editor.org/rfc/rfc6447.txt">
    RFC 6447</a>.</p>
</body>
</html>
END
```

6.2. Schema Registration for location-filter

This specification registers a schema, as per the guidelines in [RFC3688].

URI: urn:ietf:params:xml:schema:location-filter

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

XML: The XML can be found as the sole content of Section 4.

7. Contributors

We would like to thank Martin Thomson and James Polk for their contributions to this document.

8. Acknowledgments

Thanks to Richard Barnes, Alissa Cooper, Randall Gellens, Carl Reed, Ben Campbell, Adam Roach, Allan Thomson, and James Winterbottom for their comments.

Furthermore, we would like to thank Alexey Melnikov for his IESG review comments.

9. References

9.1. Normative References

- [GML] OpenGIS, "Open Geography Markup Language (GML) Implementation Specification", OpenGIS OGC 02-023r4, January 2003,
<<http://www.opengis.org/techno/implementation.htm>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3265] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002.
- [RFC3856] Rosenberg, J., "A Presence Event Package for the Session Initiation Protocol (SIP)", RFC 3856, August 2004.
- [RFC4119] Peterson, J., "A Presence-based GEOPRIV Location Object Format", RFC 4119, December 2005.
- [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.

- [RFC5491] Winterbottom, J., Thomson, M., and H. Tschafenig, "GEOPRIV Presence Information Data Format Location Object (PIDF-LO) Usage Clarification, Considerations, and Recommendations", RFC 5491, March 2009.
- [RFC5962] Schulzrinne, H., Singh, V., Tschafenig, H., and M. Thomson, "Dynamic Extensions to the Presence Information Data Format Location Object (PIDF-LO)", RFC 5962, September 2010.
- [RFC6280] Barnes, R., Lepinski, M., Cooper, A., Morris, J., Tschafenig, H., and H. Schulzrinne, "An Architecture for Location and Location Privacy in Internet Applications", BCP 160, RFC 6280, July 2011.
- [RFC6446] Niemi, A., Kiss, K., and S. Loreto, "Session Initiation Protocol (SIP) Event Notification Extension for Notification Rate Control", RFC 6446, January 2012.

9.2. Informative References

- [GEO-POLICY] Schulzrinne, H., Tschafenig, H., Cuellar, J., Polk, J., Morris, J., and M. Thomson, "Geolocation Policy: A Document Format for Expressing Privacy Preferences for Location Information", Work in Progress, October 2011.
- [HELD] Winterbottom, J., Tschafenig, H., and M. Thomson, "Location URI Contexts in HTTP-Enabled Location Delivery (HELD)", Work in Progress, October 2009.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, January 2004.
- [RFC4745] Schulzrinne, H., Tschafenig, H., Morris, J., Cuellar, J., Polk, J., and J. Rosenberg, "Common Policy: A Document Format for Expressing Privacy Preferences", RFC 4745, February 2007.
- [RFC5985] Barnes, M., "HTTP-Enabled Location Delivery (HELD)", RFC 5985, September 2010.
- [SIP-LOC] Polk, J., Rosen, B., and J. Peterson, "Location Conveyance for the Session Initiation Protocol", Work in Progress, September 2011.

Authors' Addresses

**Rohan Mahy
Individual**

EMail: rohan@ekabal.com

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

**Phone: +1 724 382 1051
EMail: br@brianrosen.net**

**Hannes Tschofenig
Nokia Siemens Networks
Linnoitustie 6
Espoo 02600
Finland**

**Phone: +358 (50) 4871445
EMail: Hannes.Tschofenig@gmx.net
URI: <http://www.tschofenig.priv.at>**