SIP Working Group
Internet Draft
Expiration: May 16th, 2008
Intended Status: Standards Track (PS)

James Polk
Cisco Systems
Brian Rosen
NeuStar

# Location Conveyance for the Session Initiation Protocol

draft-ietf-sip-location-conveyance-09.txt

Nov 16th, 2007

## Status of this Memo



## Copyright Notice



## Abstract

This document defines an extension to the Session Initiation Protocol (SIP) to convey geographic location information from one SIP entity to another SIP entity. The extension covers end to end conveyance as well as location-based routing, where proxy servers make routing decisions based on the location of the SIP user agents.

Table of Contents



## 1. Introduction

This document describes how Location can be "conveyed" (that is, transmitted over the Internet) from one SIP user agent (UA), or in some circumstances, a proxy server acting in support of a UA, to another entity using SIP [RFC3261]. Here "Location" is a description of the physical geographical area where something currently exists. The phrase "location conveyance" describes scenarios in which a SIP user agent client (UAC) is informing a user agent server (UAS), or intermediate SIP server where the UAC is. A superset of this can also be true as well, in which one UA(1) is telling another UA(2) where another Target is, meaning not necessarily where UA(1) is. The key to this is whether UA(1) has

permission to retransmit that other Target's location. If yes, then this is valid. If no, then this is breaking a fundamental rule within this extension.

Location Conveyance is different from a UAC seeking the location the UAS. Location conveyance is a 'sending location out in the request' model, where 'asking that someone else's location be in a response' is not discussed here.

Geographic location in the IETF is discussed in RFC 3693 (Geopriv Requirements) [RFC3693]. It defines a "Target" as the entity whose location is being sought. In this case, this is the UA's (UA) location. A [RFC3693] "Using Protocol" defines how a "location Server" transmits a "Location Object" to a "Location Recipient" while maintaining the contained privacy intentions of the Target intact. This document describes the extension to SIP for how it complies with the Using Protocol requirements, where the location server is a UA or Proxy Server and the Location Recipient is another UA or Proxy Server.

Location can be transmitted by-value or by-reference. The location "value" in this SIP extension is in the form of a Presence Information Data Format - Location Object, or PIDF-LO, as described in [RFC4119]. A PIDF-LO is an XML Scheme specifically for carrying geographic location of a Target. Location-by-value refers to a UA including a PIDF-LO as a body part of a SIP message, sending that Location Object to another SIP element. Location-by-reference refers to a UA or proxy server including a URI in a SIP message header field which can be dereferenced by a Location Recipient for a Location Object, in the form of a PIDF-LO. Dereferencing can be by a SIP UA or a SIP server.

As recited in RFC 3693, location often must be kept private. The Location Object (PIDF-LO) contains rules which provides guidance to the Location Recipient and controls onward distribution and retention of the location. This document describes the security and privacy considerations that must be applied to location conveyed with SIP.

Another use for location is location-based routing of a SIP request, where the choice of the next hop (and usually, the outgoing Request-URI) is determined by the location of the UAC which is in the message by-value or by-reference. This document describes how location can be conveyed from the UAC, or a proxy acting on its behalf, to a routing proxy. How the location is actually used to determine the next hop or Request-URI is beyond the scope of this document.

We refer to the "emergency case". This refers to a specific, important use of SIP location conveyance where the location of the caller is used to determine which Public Safety Answering Point (PSAP) is expected to receive an emergency call request for help

(e.g., a call to 1-1-2 or 9-1-1). This is an example of location-based routing. The location conveyed is also used by the PSAP to dispatch first responders to the caller's location. There

are special security considerations, which make the emergency case unique, compared to a normal location conveyance within SIP.

## 2. Conventions used in this document

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].

## 3. Mechanisms

### 3.1 Overview of SIP Location Conveyance

This document defines a new SIP header: Geolocation. The Geolocation header field contains a URI which can either be a "cid:" URI (Content Identification), per [RFC2392], or a location-by-reference URI to be dereferenced by a Location Recipient to retrieve the location of the Target UA.

Where the Geolocation header contains a "cid:", the URI points to a message body that is in the form of a PIDF [RFC3863], which was extended in [RFC4119] to include location, as a PIDF-LO. This is location-by-value, the actual location information in the PIDF-LO is included in the body of the message.

If the URI in the Geolocation header field is a scheme other than "cid:", another protocol operation is needed by the SIP message recipient to obtain the location of the Target (UA). This is location-by-reference. This document describes how a SIP presence subscription [RFC3856] can be used as a dereference protocol.

The Geolocation header, either with the PIDF-LO in a body or as a location-by-reference URI, can be included by a UA in a SIP message. A SIP proxy server may assert location of the UA by inserting the header field, which must specify a location-by-reference URI. Since body parts cannot be inserted by a SIP proxy server, location-by-value message body cannot be inserted by a proxy.

The Geolocation header can have parameters that are associated with a URI in the header field. The "inserted-by" parameter has values of "endpoint" or "server", indicating which entry added location to the message. This header parameter MAY be added every time a new location is added into a message.

Retargeting means the Request-URI of the request has changed to point at a new destination UAS. This is different than message routing, that all SIP proxies do. If a SIP request is retargeted

based on the location contained or referenced within that message, the "used-for-routing" parameter MUST be added as a header parameter

within the appropriate locationValue.

There is no mechanism by which the veracity of these parameters can be verified. They are hints to downstream entities on how the location information in the message was originated and used.

This document creates a new option tag: geolocation, to indicate support for the this extension by UAs.

A new error message (424 Bad Location Information) is also defined in this document. Within this response is a new header indicating location-based errors, call the Geolocation-Error header. This header has various codes that provide additional information about the type of location error experienced by a Location Recipient.

Both new headers, the header parameters, the new option-tag, the new error response, and Geolocation-Error codes are IANA registered by this document.

### 3.2 The Geolocation Header

This document defines and IANA registers a new SIP header: Geolocation. The Geolocation header field MUST contain at least one of two types of URIs:

- a location-by-reference URI, or
- a content-ID indicating where location is within the message body of this message

A location-by-reference URI is a pointer to a record on a remote node containing location of the location Target, typically the UA in this transaction.

A location-by-value content-ID (cid-url) [RFC2392] indicates which message body part contains location for this UA.

The Geolocation header has the following BNF syntax:

```
Geolocation        =  "Geolocation" HCOLON (locationValue *(COMMA
                       locationValue))
locationValue      =  LAQUOT locationURI RAQUOT *(SEMI geoloc-param)
locationURI        =  sip-URI / sips-URI / pres-URI
                       / cid-url ; (from RFC 2392)
                       / absoluteURI ; (from RFC 3261)
geoloc-param       =  "inserted-by" EQUAL geoloc-inserter
                       / "used-for-routing"
                       / "recipient" EQUAL recipient-type
                       / generic-param ; (from RFC 3261)
```

```
   geoloc-inserter    =  host-id
                          / gen-value ; (from RFC 3261)
```

recipient-type = "endpoint" / "routing-entity" / "both"
/ gen-value ; (from RFC 3261)

sip-URI, sips-URI and absoluteURI are defined according to RFC 3261. The pres-URI is defined in RFC 3859 [RFC3859].

The cid-url is defined in [RFC2392] to locate message body parts. This URI type MUST be present in a SIP message if location is by-value in that same message.

Other protocols used in the Location URI MUST be reviewed against the RFC 3693 criteria for a Using Protocol.

The Geolocation header MAY have one or more locationValues. SIP servers inserting a locationValue MUST add the new value to the end of the header value, such that the last locationValue in the header is the most recent one added to the message.

A locationValue has the following independent header parameters,

- the "inserted-by=" parameter provides the host-id (alice.example.com -- which is the same as the "sent-by" parameter in a Via header) of the SIP entity that inserted this locationValue into the request. This is used to map to any Geolocation-Error message to determine which location, if there is more than one in a request, the error corresponds to. If an entity receives an Geolocation-Error with a host-id not of this entity, the Geolocation-Error SHOULD be ignored.

- the "used-for-routing" parameter to inform recipients that the location in this locationValue was used to route the message towards the ultimate destination UAS. This can occur more than once along the request's path. Because locationValues are inserted as last inserted is last in the header, the last locationValue is the most recent one added to the message. This also gives the "used-for-routing" header parameter added integrity - as the receiving SIP entity knows which locationURI the message was routed upon.

- the "recipient=" parameter to allow recipients to infer what SIP element type this locationValue was intended to be for. The types are

    - "endpoint" - meaning the ultimate destination UAS;

    - "routing-entity" - meaning SIP servers that route messages based on the location contents of requests; and

    - "both" - meaning this locationValue is to be viewed by both types of SIP entities.

Not all SIP entities have to read the locationValue within a

Geolocation header, therefore a parameter value of "both" does not mean "every" SIP element receiving this request, it means all that care to pay attention to a locationValue. The default behavior of SIP entities reading the locationValue is that if this header parameter is NOT present, the intended recipient is the destination UAS.

Each locationValue MUST contain exactly one "inserted-by" parameter, indicating which SIP entity added the locationValue to the SIP request.

Each of the three types of header parameters listed here MAY appear in any locationValue once. There MUST NOT be more than one "inserted-by=" parameter or one "used-for-routing" parameter or one "recipient=" parameter in the same locationValue. However, there can be more than one locationValue in the same Geolocation header.

This document defines the Geolocation header as valid in the following SIP requests:

| | |
|---|---|
| INVITE [RFC3261], | REGISTER [RFC3261], |
| OPTIONS [RFC3261], | BYE [RFC3261], |
| UPDATE [RFC3311], | INFO [RFC2976], |
| MESSAGE [RFC3428], | REFER [RFC3515], |
| SUBSCRIBE [RFC3265], | NOTIFY [RFC3265], |
| PUBLISH [RFC3903] and | PRACK [RFC3262] |

Discussing location using the PUBLISH Request Method is out of scope for this document, but the Table 1 shows PUBLISH is to support Location Conveyance via this extension.

The following table extends the values in Table 2&3 of RFC 3261 [RFC3261].

| Header field | where | proxy | INV | ACK | CAN | BYE | REG | OPT | PRA |
|---|---|---|---|---|---|---|---|---|---|
| Geolocation | R | ar | o | - | - | o | o | o | o |

| Header field | where | proxy | SUB | NOT | UPD | MSG | REF | INF | PUB |
|---|---|---|---|---|---|---|---|---|---|
| Geolocation | R | ar | o | o | o | o | o | o | o |

Table 1: Summary of the Geolocation Header

The Geolocation header field MAY be included in any one of the above requests by a UAC. A proxy MAY add the Geolocation header, but MUST NOT modify any pre-existing locationValue, including its associated header parameters of within an existing Geolocation header value,

unless one of the existing locationValues is used to retarget the request towards a new destination UAS. This is discussed in section

5.3.

[RFC3261] states message bodies cannot be added by proxies. Therefore, any Geolocation header field added by a proxy MUST be in the form of a location-by-reference URI, in its own locationValue header value.

Adding a new locationValue to an existing Geolocation header SHOULD NOT occur without appropriate caution to the fact that Location Recipients might not understand how to process more than one location, given this document's limited guidance as to what a Location Recipient should do when receiving more than one location (i.e., currently no priority instructions are given for which locationValue to use if there are more than one). A Location Recipient can easily be confused by too much location information, producing undesirable results. The <tuple id> element in the PIDF-LO XML indicates whose location is contained in the PIDF-LO.

Location Recipients receiving a location object, received directly or as the result of a dereference, MUST honor the usage element rules within that XML document, per RFC 4119. Such entities MUST NOT alter the rule set.

## 3.3 424 (Bad Location Information) Response Code

If a UAS or SIP intermediary detects an error in a request message specific to the location information supplied by-value or by-reference. The new 4XX level error is created here to indicate a problem with the location in the request message. This document creates and IANA registers the new error code:

424 (Bad Location Information)

The 424 (Bad Location Information) response code is a rejection of the request, due to its location contents, indicating the location information was malformed or not satisfactory for the recipient's purpose, or could not be dereferenced.

Section 3.4 creates the Geolocation-Error header to provide more detail about what was wrong with the location information in the request. This header MUST be in the 424 response, containing a locationErrorValue for each invalid locationValue in the request (i.e., and one-for-one matching if all locationValues in the request were bad).

If more than one location is present in a request (by-value or by-reference), and any of the locationValues is good for the Location Recipient to process, a 424 MUST NOT be sent. The 424 is only appropriate when the Location Recipient needs a locationValue

and there are no locationValues included in a SIP request that are usable by a recipient.

A 424 (Bad Location Information) response is a final response within a transaction, and does not terminate a dialog.

The UAC can use whatever means it knows of to verify/refresh its location information before attempting a new request that includes location. There is no cross-transaction awareness expected by either the UAS or SIP intermediary as a result of this error message.

The new 424 (Bad Location Information) error code is IANA registered in Section 8 of this document. An initial set of location error of IANA registered Geolocation-Error codes are in Section 3.4 of this document.

### 3.4 The Geolocation-Error Header Providing Error Granularity

As discussed in Section 3.3, more granular error notifications, specific to location errors within a received request, are required if the UAC is to know what was wrong within the original request. The Geolocation-Error header is created here for this purpose. Geolocation-Error header is used to convey location specific errors within a response. Additions to this IANA registered header require an RFC be published.

```
Geolocation-Error        = "Geolocation-Error" HCOLON
                           [locationErrorValue
                           *(COMMA locationErrorValue)]
locationErrorValue       = location-error-code *(SEMI
                           location-error-params)
location-error-code      = 1*3DIGIT
location-error-params    = location-error-node-id
                         / DQOUTE location-error-host-id DQOUTE
                         / CAtype *(SEMI CAtype)
                         / DQOUTE location-error-code-text DQOUTE
                         / generic-param ; from RFC3261
location-error-node-id   = "node" EQUAL hostname; from RFC3261
location-error-host-id   = "inserter" EQUAL hostname ; from RFC3261
CAtype                   = "CAtype" EQUAL civic-code *(SEMI "CAtype"
                             EQUAL civic-code)
location-error-code-text = "code" EQUAL quoted-string ; from RFC3261
civic-code               = IANA registered CAtypes; from
                                                [IANA-civic]
```

The Geolocation-Error header MUST contain at least one locationErrorValue to indicate what was wrong with the original locationValue in the corresponding request. If a Location Recipient experienced more than one error in the locationValue of the corresponding SIP request, there can be one locationErrorValue per

problem with the locationValue in the request (the except to this is involving CAtypes, which will be covered later here). If there was something wrong with more than one locationValue in a request, a

corresponding locationErrorValue would be sent, one per error, in the response. Each locationErrorValue contains a 3-digit error code (defined in subsections to this section of this doc) indicating what was wrong with the location(s) in the request. Each error type has a corresponding quoted error text string that is human understandable.

Also within the locationErrorValue is the Location Recipient identifier (the "node=") who experienced the location error, as well as an identifier of which SIP entity (the "inserter=") the Location Recipient is told (in the locationValue) added the locationValue to this request. The "node=" and "inserter=" are domain identifier of a SIP entity, the same as is entered in the "sent-by" parameter of the Via header for that entity [RFC3261]. As stated in section 18 of RFC 3261, the usage of FQDN is RECOMMENDED. Here are examples of both

node=bob.example.com

inserter=alice.example.com

Both "node=" and "inserter=" parameters MUST be present in all locationErrorValues in a response, unless the "inserted-by=" parameter was not in the request. The "inserter=" parameter is copied from the "inserted-by=" parameter within the locationValue of the request.

Here's why, a Location Recipient that experienced the location problem with the request needs to tell who added which location into the original request. Since more than one SIP entity can insert location into a request, all other SIP elements may be confused by receiving this error header. So, the header has to identify who it is for, so that all other SIP entities that read the header know to ignore it, since it is not for them. This is of particular use if the original UAC did not include a locationValue in the original SIP request, but a SIP server along the path did insert a locationValue. The locationErrorValue would travel to each SIP entity along the original path and tell both the server that included the locationValue what was wrong with the location and the UAC who did not know what the error meant.

A worse case is when both the original UAC and a SIP server along the path included a locationValue, but there was only something wrong with one of the locationValues. Without this identification of which locationValue was in error, both entities would react and one would do so incorrectly.

Finally, there can be a list of one or more CAtype civic-codes that are determined to be in error by the Location Recipient. Perhaps

the Location Recipient believes one or more CAtypes are missing, and required in order to fully process the locationValue in the request, or perhaps data entered in one or more CAtypes is wrong, according

to the Location Recipient. The list of CAtypes is taken from those that are IANA registered at [IANA-civic].

More than one locationErrorValue in a Geolocation-Error header is separated by a comma.

If more than one locationErrorValue is in a response, and intended for the same "inserter=", the error codes SHOULD NOT conflict in meaning. In other words, two error codes (within separate locationErrorValues of the same response) SHOULD NOT give misleading or inconsistent indications to the location "inserter=".

Here is an example of a Geolocation-Error header

```
Geolocation-Error: 106; "node=bob.example.com";
                        "inserter=alice.example.com";
                        CAtype=A3; CAtype=STS;
                        code="incomplete location supplied"
```

In this example, the Location Recipient (node=Bob) has determined the location supplied by the "inserter=" (inserter=Alice) was not enough to determine where (Alice) was. Specifically, Bob has determined that CAtypes A3 (the city) and STS (the street type (Street, Road, Avenue, etc)) were not present to form a complete location of the Alice. A subsequent request by Alice that included these two additional pieces of location information would tell Bob where Alice was.

Notice the CAtypes that were in error are (in the above example Geolocation-Error header), according to the Location Recipient, listed in the locationErrorValue. The associated CAtype values MUST NOT be listed in the locationErrorValue. This is for privacy/security concerns. It is up to the Location Recipient to determine which CAtypes were in error, and only list those CAtypes in the response. The "inserter=" entity MUST determine what to do about correcting each CAtype found in error for subsequent location conveyance. Usually, this would involve either refreshing its location information however it learned its location in the first place, or merely listing what information is lacking/wrong to the location sender (i.e., the user) or its network management.

The following table extends the values in Table 2&3 of RFC 3261 [RFC3261].

| Header field | where | proxy | INV | ACK | CAN | BYE | REG | OPT | PRA |
|---|---|---|---|---|---|---|---|---|---|
| Geolocation-Error | r | ar | o | - | - | o | o | o | o |

| Header field | where | proxy | SUB | NOT | UPD | MSG | REF | INF | PUB |
|---|---|---|---|---|---|---|---|---|---|

Geolocation-Error r ar o o o o o o o

Table 2: Summary of the Geolocation-Error Header

The Geolocation-Error header field MAY be included in any response to one of the above SIP requests, so long as Geolocation was in the request part of the transaction. The choice of which SIP requests are in table 2 above come from which Methods can optionally have the Geolocation header (see section 3.2).

Here is an example of a transaction that has a location error. In this case, Bob responds with a 424 (Bad Location Information) response, including a Geolocation-Error header, is in Figure 1.

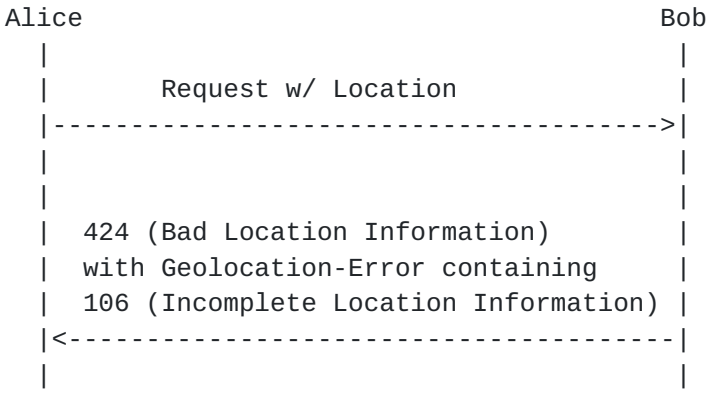


Figure 1. Basic Transaction with 424 and Geolocation-Error Header

The following subsections provide an initial list of location specific granular codes for any SIP responses, including the new 424 (Bad Location Information) response. If more than one specific Geolocation-Error code is applicable for a response, each MUST be in the response. Geolocation-Error Code 100 is the generic 'location was supplied, but not understood' error. If a more specific code applies, a code 100 is unnecessary.

### 3.4.1 Geolocation-Error Code 100 Location Not Understood

Geolocation-Error code 100 "Location Format not supported" means the location format supplied in the request, by-value or by-reference, was not supported.

This code means the recipient understood that location was included in the message, but the format is not supported. Perhaps the format was a freeform text format or data-URL and the recipient only understood location in RFC 4119 PIDF-LO format (civic or x.y(.z) coordinate). This error code applies when a recipient has difficulty parsing the location supplied in the request.

If the format is understood, but not desired, an error code 101 or 102 MUST be returned in a 424 response, depending on which location

format is desired. The Location Recipient returns an error code 101 or 102 when it only understands one location format (coordinate or

civic) and did not receive that format.

If a more specific error code is appropriate in a response, including error code 100 is unnecessary.

error-text-string: Location format not supported

An example usage in a SIP 424 response:

```
Geolocation-Error: 100; "node=bob.example.com";
                        "inserter=alice.example.com";
                        CAtype=A3; CAtype=STS;
                        code="Location Format not supported"
```

### 3.4.2 Geolocation-Error Code 101 Coordinate-location Format Desired

Geolocation-Error code 101 "Coordinate-location Format Desired" means the location format supplied in the request (probably formatted in civic), by-value or by-reference, was understood and supported, but that the recipient, or an application on the recipient, can or prefers to only process location in the coordinate-location format.

A typical reaction to receiving this code is to resend the original message with location formatted in coordinate instead.

error-text-string: Coordinate-location Format Desired

An example usage in a SIP 424 response:

```
Geolocation-Error: 101; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="Coordinate-location Format Desired"
```

### 3.4.3 Geolocation-Error Code 102 Civic-location Format Desired

Geolocation-Error code 102 "Civic-location Format Desired" means the location format supplied in the request (probably formatted in coordinate), by-value or by-reference, was understood and supported, but that the recipient, or an application on the recipient, can or prefers to only process location in the civic-location format.

A typical reaction to receiving this code is to resend the original message with location formatted in civic instead.

error-text-string: Civic-location Format Desired

An example usage in a SIP 424 response:

```
Geolocation-Error: 102; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="Civic-location Format Desired"
```

### 3.4.4 Geolocation-Error Code 103 Cannot Parse Location Supplied

Geolocation-Error code 103 "Cannot parse location supplied" means the location provided, whether by-value or by-reference, in a request is not well formed.

error-text-string: Cannot parse location supplied

An example usage in a SIP 424 response:

```
Geolocation-Error: 103; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="Cannot parse location supplied"
```

### 3.4.5 Geolocation-Error Code 104 Cannot Find Location Information

Geolocation-Error code 104 "Cannot find location" means location was expected in the request, but the recipient cannot find it.

This can be either because the cid: pointed to a message body part that is not present in the request, there was no location message body part, or what is dereferenced at the supplied locationURI did not return a PIDF-LO, or location is encrypted/opaque to the recipient.

A typical reaction to receiving this code is for the location sender to verify that it has indeed included location information in the request in the properly indicated place and then to send the request again.

error-text-string: Cannot find location

An example usage in a SIP 424 response:

```
Geolocation-Error: 104; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="Cannot find location"
```

### 3.4.6 Geolocation-Error Code 105 Conflicting Locations Supplied

Geolocation-Error code 105 "Conflicting Locations Supplied" means a Location Recipient received more than one location describing where the Target is, and is either unsure which whole location is true or which parts of multiple locations make up where the Target is.

This is generally a case of either too much information, and the

information is pointing towards at least two different positions, confusing the recipient.

A possible scenario exists in which at least two locations are in the request, perhaps one or more were added by proxies along the path of the request, each pointing to where the UAC is. If these are pointing at different positions - the UAS does not know which to trust. This error code unfortunately means the UAS cannot solve for which location needs to be ignored to make up a complete location, or how to prioritize one location over all others in the same request.

A typical reaction to receiving this code is to reduce the number of different locations supplied in the request, if under control by the Target, and send another message to the Location Recipient.

error-text-string: Conflicting Locations Supplied

An example usage in a SIP 424 response:

```
Geolocation-Error: 105; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="Conflicting Locations Supplied"
```

### 3.4.7 Geolocation-Error Code 106 Incomplete Location Supplied

Geolocation-Error code 106 "Incomplete Location Supplied" means there is not enough location information, by-value or retrieved by-reference, to determine where the location Target is.

Perhaps the coordinate precision is not fine enough, or the civic address lacks the fields to inform the UAS or proxy where the Target is. This might be true for request retargeting, or it might be true for first responder dispatch or pizza delivery (for example, because the street address is missing).

A typical reaction to receiving this code is for the location sender to convey more (precise) location information, if doing so is allowed by local policy.

error-text-string: Incomplete location supplied

An example usage in a SIP 424 response:

```
Geolocation-Error: 106; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="Incomplete location supplied"
```

### 3.4.8 Geolocation-Error Code 107 Cannot Dereference

Geolocation-Error code 107 "Cannot dereference" means the act of dereferencing failed to return the Target's location. This generally means the supplied URI is bad.

error-text-string: Cannot dereference

An example usage in a SIP 424 response:

```
Geolocation-Error: 107; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="Cannot dereference"
```

### 3.4.9 Geolocation-Error Code 108 Dereference Denied

Geolocation-Error code 108 "Dereference Denied" means there was insufficient authorization to dereference the Target's location.

error-text-string: Dereference Denied

An example usage in a SIP 424 response:

```
Geolocation-Error: 108; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="Dereference Denied"
```

### 3.4.10 Geolocation-Error Code 109 Dereference Timeout

Geolocation-Error code 109 "Dereference Timeout" means the dereferencing node has not received the Target's location within a reasonable timeframe.

error-text-string: Dereference Timeout

An example usage in a SIP 424 response:

```
Geolocation-Error: 109; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="Dereference Timeout"
```

### 3.4.11 Geolocation-Error Code 110 Cannot Process Dereference

Geolocation-Error code 110 "Cannot process Dereference" means the dereference protocol has received an overload condition error, indicating the location cannot be accessed at this time.

If a SIP or SIPS scheme were used to dereference the Target's

location, and a 503 (Service Unavailable) were the response to the

dereference query, this Geolocation-Error code 11 would be placed in the 424 (Bad Location Information) response to the location sender.

error-text-string: Cannot process Dereference

An example usage in a SIP 424 response:

```
Geolocation-Error: 110; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="Cannot process Dereference"
```

### 3.4.12 Geolocation-Error Code 120 Unsupported Scheme - SIP desired

Geolocation-Error code 120 "Unsupported Scheme - SIP desired" means the location dereferencer cannot dereference using the location-by-reference URI scheme supplied because it does not support the necessary protocol to do this.

This code means the Location Recipient can dereference the Target's location using a SIP-URI scheme. There can be more than one locationErrorValue in a Geolocation-Error header, indicating in this context the recipient can dereference using each scheme protocol included in the Geolocation-Error header.

Note that indicating SIP to be used to dereference location is requesting the transmission to be in cleartext, which is a security risk. Therefore, the SIP scheme SHOULD NOT be used to dereference. An exception can be made for emergency calling, preferably after SIPS has been attempted, and failed.

A typical reaction to receiving this code would be for the location sender to send a URI with the sip scheme.

error-text-string: unsupported scheme - SIP desired

An example usage in a SIP 424 response:

```
Geolocation-Error: 120; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="unsupported scheme - SIP desired"
```

### 3.4.13 Geolocation-Error Code 121 Unsupported Scheme - SIPS desired

Geolocation-Error code 121 "Unsupported Scheme - SIPS desired" means the location dereferencer cannot dereference using the location-by-reference URI scheme supplied because it does not support the necessary protocol to do this.

This code means the Location Recipient can dereference the Target's location using a SIPS-URI scheme. There can be more than one

locationErrorValue in a Geolocation-Error header, indicating in this context the recipient can dereference using each scheme protocol included in the Geolocation-Error header.

error-text-string: unsupported scheme - SIPS desired

An example usage in a SIP 424 response:

```
Geolocation-Error: 121; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="unsupported scheme - SIPS desired"
```

### 3.4.14 Geolocation-Error Code 122 Unsupported Scheme - pres desired

Geolocation-Error code 122 "Unsupported Scheme - pres desired" means the location dereferencer cannot dereference using the location-by-reference URI scheme supplied because it does not support the necessary protocol to do this.

This code means the Location Recipient can dereference the Target's location using a PRES-URI scheme. There can be more than one locationErrorValue in a Geolocation-Error header, indicating in this context the recipient can dereference using each scheme protocol included in the Geolocation-Error header.

error-text-string: unsupported scheme - pres desired

An example usage in a SIP 424 response:

```
Geolocation-Error: 122; "node=bob.example.com";
                        "inserter=alice.example.com";
                        code="unsupported scheme - pres desired"
```

## 3.5 The Geolocation Option Tag

This document creates and IANA registers one new option tag: "geolocation". This option tag is to be used, per RFC 3261, in the Require, Supported and Unsupported headers. Whenever a UA wants to indicate support for this SIP extension, the geolocation option tag is included in a Supported header of the SIP message.

Including the geolocation option-tag within an Unsupported header of a 420 (Bad Extension) response is appropriate when a UAS does not support this Geolocation extension.

A UAC adding this option-tag to a Require header field indicates to a UAS the UAS MUST support this extension in order to continue processing the message, or send a 420 response back to the UAC.

Some environments might use a Require header in this way, but it should be used with caution to prevent unnecessary communications

problems.

A UAC SHOULD NOT include this option tag in a Proxy-Require header, since a UAC is not likely to understand the topology of the infrastructure, and therefore would not understand which proxy will do the location-based routing function, if any. A potentially bad scenario would have the first proxy not support this extension, but a subsequent proxy does. This would result in no communications past the first proxy, which MUST send the 420 back under these circumstances.

### 3.6 Using sip/sips/pres as a Dereference Scheme

If a location-by-reference (LbyR) URI is included in a SIP request, it MUST be in a locationValue in the Geolocation header and it MUST be a SIP, SIPS or PRES-URI . When PRES: is used, if the resulting resolution, per [RFC3856], resolves to a SIP: or SIPS: URI, this section applies. Use of other protocols for dereferencing of a PRES: URI is not defined, and such use is subject to review against RFC 3693 Using Protocol criteria.

Dereferencing a Target's location using SIP or SIPS MUST be accomplished by treating the URI as a presence URI and generating a SUBSCRIBE request to a presence server as per [RFC3856] using the 'presence' event package. The resulting NOTIFY will contain a PIDF, which MUST contain a PIDF-LO. See Figure 2. for a basic message flow for a dereference.

When used in this manner, SIP is a Using Protocol per [RFC3693] and elements receiving location MUST honor the 'usage-element' rules as defined in this extension.

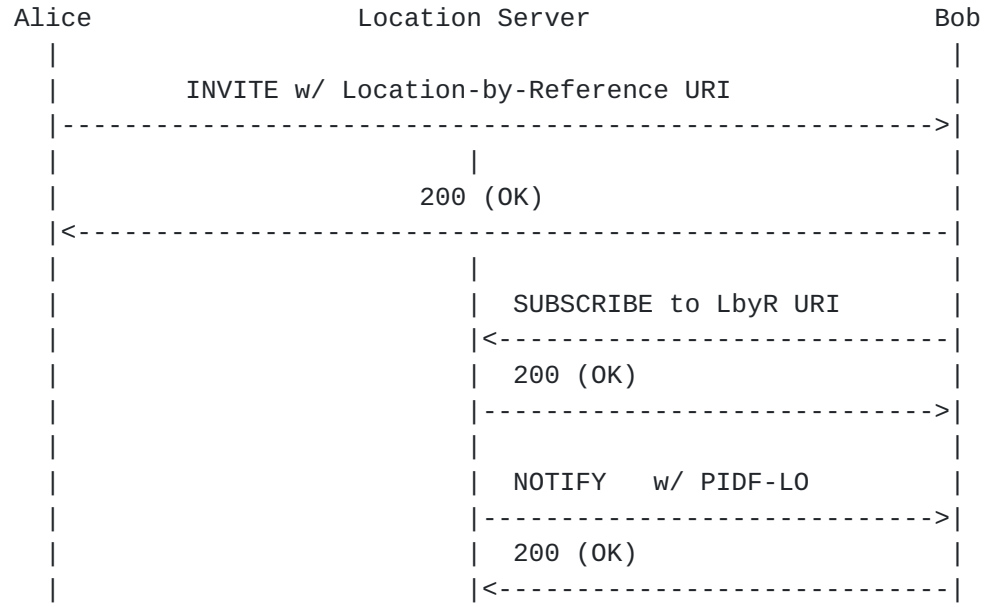

Figure 2. Location-by-Reference and Dereferencing

In Figure 2., Alice sends Bob her location in a LbyR URI. Bob receives this LbyR URI in the INVITE and generates a new transaction (SUBSCRIBE) to retrieve the PIDF-LO of Alice. If accepted, the PIDF-LO will be in the NOTIFY request from the Location Server. This is the first instance between Alice and Bob that Alice's location is in any message, therefore it is sent only once, from the Location Server to Bob.

A dereference of a location-by-reference URI using SUBSCRIBE is not violating a PIDF-LO 'retransmission-allowed' element value set to 'no', as the NOTIFY is the only message in this multi-message set of transactions that contains the Target's location, with the location recipient being the only SIP element to receive location - which is the purpose of this extension: to convey location to a specific destination.

## 4. Geolocation Examples

This section contains are two examples of messages providing location. One shows location-by-value with coordinates, the other shows location-by-reference. The example for (Coordinate format) is taken from [RFC3825]. A civic format example of the same position on the earth as is in the coordinate format example is in appendix B, which is taken from [RFC4776]. The differences between the two formats are within the <gp:location-info> of the examples. Other than this portion of each PIDF-LO, the rest is the same for both location formats.

The key to the provided samples is in the Geolocation header, which has a different type of URI, based on the different means of location conveyance. Section 4.1 shows a "cid:" URI, indicating this SIP request contains a location-by-value message body - which is in the form of a PIDF-LO. Section 4.2 shows a location-by-reference URI indicating location is to be acquired via an indirection dereference mechanism, which is determined by the scheme of URI supplied.

### 4.1 Location-by-value (Coordinate Format)

This example shows an INVITE message with a coordinate, or coordinate location. In this example, the SIP request uses a sips-URI [RFC3261], meaning this message is TLS protected on a hop-by-hop basis all the way to Bob's domain.

```
INVITE sips:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/TLS pc33.atlanta.example.com;branch=z9hG4bK74bf9
Max-Forwards: 70
```

```
   To: Bob <sips:bob@biloxi.example.com>
   From: Alice <sips:alice@atlanta.example.com>;tag=9fxced76sl
```

```
   Call-ID: 384827629822018851l@atlanta.example.com
   Geolocation: <cid:target123@atlanta.example.com>
     ;inserted-by=alice@atlanta.example.com ;recipient=endpoint
   Supported: geolocation
   Accept: application/sdp, application/pidf+xml
   CSeq: 31862 INVITE
   Contact: <sips:alice@atlanta.example.com>
   Content-Type: multipart/mixed; boundary=boundary1
   Content-Length: ...

   --boundary1

   Content-Type: application/sdp

   ...SDP goes here

   --boundary1

   Content-Type: application/pidf+xml
   Content-ID: alice123@atlanta.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:cl="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
          entity="pres:alice@atlanta.example.com">
        <tuple id="sg89ae">
         <timestamp>2007-12-02T14:00:00Z</timestamp>
         <status>
          <gp:geopriv>
            <gp:location-info>
              <gml:location>
                <gml:Point srsName="urn:ogc:def:crs:EPSG::4326">
                  <gml:pos>33.001111 -96.68142</gml:pos>
                </gml:Point>
               </gml:location>
            </gp:location-info>
            <gp:usage-rules>
              <gp:retransmission-allowed>no</gp:retransmission-allowed>
              <gp:retention-expiry>2007-12-07T18:00:00Z</gp:retention-
                             expiry>
            </gp:usage-rules>
            <gp:method>DHCP</gp:method>
            <gp:provided-by>www.example.com</gp:provided-by>
          </gp:geopriv>
         </status>
        </tuple>
       </presence>
   --boundary1--
```

The Geolocation header field from the above INVITE...

Geolocation: <cid:target123@atlanta.example.com>

...indicates the content-ID location [RFC2392] within the multipart message body of where location information is, with SDP being the other message body part.

If the Geolocation header field were this instead:

Geolocation: <sips:server5.atlanta.example.com/target123>

...this would indicate location by-reference was included in this message. It is expected that any node wanting to know where user target123 is would subscribe to server5 to dereference the sips-URI. The returning NOTIFY would contain Alice's location in a PIDF-LO, as if it were included in a message body (part) of the original INVITE here.

### 4.2 Location-by-reference

Below is an INVITE request with a location-by-reference URI instead of a location-by-value PIDF-LO message body part shown in Sections 4.1. It is up to the location recipient to dereference Alice's location at the Atlanta server containing the location record. Dereferencing, if done with SIP, is accomplished by the Location Recipient sending a SUBSCRIBE request to the URI reference for Alice's location. The received NOTIFY is the first SIP message containing Alice's UA location, as a PIDF-LO message body. The NOTIFY, in this case, is the SIP request that is conveying location, and not the INVITE. There is no retransmission of location in this usage.

```
INVITE sips:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/TLS pc33.atlanta.example.com
  ;branch=z9hG4bK74bf9
Max-Forwards: 70
To: Bob <sips:bob@biloxi.example.com>
From: Alice <sips:alice@atlanta.example.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@atlanta.example.com
Geolocation: <sips:3sdefrhy2jj7@lis.atlanta.example.com>
  ;inserted-by=bigbox3.atlanta.example.com ;recipient=both
Supported: geolocation
Accept: application/sdp, application/pidf+xml
CSeq: 31862 INVITE
Contact: <sips:alice@pc33.atlanta.example.com>

...SDP goes here as the only message body
```

A Location Recipient would need to dereference the sips-URI in the Geolocation header field to retrieve Alice's location. If the

atlanta.example.com domain chooses to implement location conveyance and delivery in this fashion (i.e., location-by-reference), it is

RECOMMENDED that entities outside this domain be able to reach the dereference server, otherwise this model of implementation is only viable within the atlanta.example.com domain.

## 5. SIP Element Behavior

Because a device's location is generally considered to be sensitive in nature, privacy of the location information needs to be protected when transmitting such information. Section 26 of [RFC3261] defines the security functionality SIPS for transporting SIP messages with either TLS or IPSec, and S/MIME for encrypting message bodies from SIP intermediaries that would otherwise have access to reading the clear-text bodies. SIP endpoints SHOULD implement S/MIME to encrypt the PIDF-LO message body (part) end-to-end when the Location Recipient is intended to be another UA. The SIPS-URI from [RFC3261] MUST be implemented for message protection (message integrity and confidentiality) and SHOULD be used when S/MIME is not used. Possession of a dereferenceable location URI can be equivalent to possession of the location information itself and thus TLS SHOULD be used when transmitting location-by-reference hop-by-hop along the path to the Location Recipient.

A PIDF includes identity information. It is possible for the identity in the PIDF to be anonymous. Implementations of this extension should consider the appropriateness of including an anonymous identity in the location information where a real identity is not required. When using location-by-reference, it is RECOMMENDED that the URI does not contain any user identifying information (for example use 3fg5t5yqw@example.com rather than alice@example.com).

Location Recipients MUST obey the privacy and security rules in the PIDF-LO as described in RFC 4119 regarding retransmission and retention.

Self-signed certificates SHOULD NOT be used for protecting a PIDF, as the sender does not have a secure identity of the recipient.

More than one location format (civic and coordinate) MAY be included in the same message body part, but all location parts of the same PIDF-LO MUST point at the same position on the earth. The same location in multiple formats can allow the recipient to use the most convenient or preferable format for its use. Multiple PIDF-LOs are allowed in the same request, with each allowed to point at separate positions - because each PIDF-LO has a Target identifier in it. Therefore, there will be no confusion by a Location Recipient receiving more than one PIDF-LO (in a message body or when dereferenced, or a combination).

It is RECOMMENDED there is only one "location" in a single SIP Request for a given Target. This means SIP servers SHOULD NOT add

another locationValue to a SIP request that already contains location. This will likely lead to confusion at the ultimate location recipient because this extension does not provide guidance on what a recipient is to do with more than one location, nor does it give any preference regarding which location is better or worse than another location in the same request.

It is allowed, but NOT RECOMMENDED, for more than one SIP element to insert location into a request along its path. As described earlier in this document, each insertion of location into a SIP request is accompanied by a locationValue in a Geolocation header. Also described earlier, each locationValue MUST contain an "inserted-by=" value indicated to a Location Recipient which host inserted location into a particular request.

### 5.1 UAC Behavior

A UAC can send location in a SIP request, because it is expected to facilitate location-based routing of the request, or spontaneously (i.e., a purpose not defined in this document but known to the UAC).

A UAC conveying location MUST include a locationValue in a Geolocation header (see section 3.2) with either a location-by-value indication (a cid-URL), or a location-by-reference indication (a dereferenceable URI). A location-by-value message body sent without a Geolocation header field MUST NOT occur. The UAC supporting this extension MUST include a Supported header with the geolocation option tag.

The geolocation option-tag is inserted in a Supported header by a UAC to provide an indication of support for this extension. The presence of the geolocation option tag in a Supported header without a Geolocation header field in the same message informs a receiving SIP element the UAC understands this extension, but it does not know or wish to convey its location at this time. Certain scenarios exist (location-based retargeting) in which location is required in a SIP request in order to retarget the message properly. This affects how a UAS or SIP server processes to such a request.

The geolocation option tag SHOULD NOT be used in the Proxy-Require Header, because the UAC often will not know the underlying topology to know which proxy will do the retargeting, thus increasing the likelihood of a request failure by the first hop proxy that does not understand this extension, but is required to by inclusion of the option-tag in this header.

A UAC inserting a locationValue MUST include an "inserted-by="

parameter to indicate its host-id. This is copied to the "inserter=" parameter of the Geolocation-Error header in a response if there is something wrong with the location in the original

request. Because more than one locationValue can be inserted along the path of the request, this indication is necessary to show which locationValue had the problem in the response. For example:

```
Geolocation: <cid:alice123@atlanta.example.com>;
              inserted-by=alice@atlanta.example.com
```

The UAC MAY include an intended target of this location parameter by adding the "recipient=" parameter to the locationValue like this:

```
Geolocation: <cid:alice123@atlanta.example.com>;
              inserted-by=alice@atlanta.example.com;
              recipient=endpoint
```

See section 3.2 for further details about all the header parameters of a locationValue.

A UAC MAY SUBSCRIBE to a LbyR URI, using the 'presence' event package, for its own location. The obvious reason for this is for the UAC to have its LbyV local to it. This document does not give a reason why a UAC would want to do this.

### 5.1.1 UAC Receiving a Location Failure Indication

If a sent request failed based on the location in the original request, a 424 (Bad Location Information) response is sent back to the UAC. The 424 MUST have a Geolocation-Error header containing one or more locationErrorValues in the response message. A locationErrorValue has a header parameter indicating which entity inserted the location pertaining to this error, called the "inserter=" parameter. This "inserter=" parameter is copied from the "inserted-by=" parameter of the locationValue by the UAS or proxy sending the error response. A UAC receiving this 424 should review this "inserter=" parameter in the locationErrorValue to see if it indicates this UAC. If locationErrorValue does not, the locationErrorValue should be ignored, and the response SHOULD be treated as a 4XX response. If locationErrorValue does indicate this UAC, this UAC MUST process the response, including the Geolocation-Error code (defined in section 3.4).

In addition to the error code, there MAY be a list of CAtypes in the locationErrorValue. If there are any, these are what the UAS or proxy determined was wrong with the location contained in the original response. The listed CAtypes will not contain the values sent by the UAC in the request. This is for security/privacy reasons.

The UAC SHOULD take correct steps to rectify future errors, based on the received error code and any CAtypes listed, to increase the

probability of successful requests in the future. A UAC MAY reattempt a new request if it believes it can correct the stated

failure in the Geolocation-Error header.

Any UAC that inserted location into a request should be prepared to receive the Geolocation-Error header in any response, looking to determine if the header is meant for the UAC, and to react accordingly.

If a UAC includes location in a request, and either the UAS does not determine errored location was critical to the transaction and accept the request, or the request failed for another reason than location, any response MAY contain a Geolocation-Error header containing a locationErrorValue with the details of the location error.

### 5.2 UAS Behavior

If the Geolocation header field is present in a received SIP request, the type of URI contained in the locationValue will indicate if location has been conveyed by-value in a message body (part) or by-reference, requiring an additional dereference transaction. If the by-reference URI is sip:, sips: or pres:, the UAS MUST initiate a SUBSCRIBE to the URI provided to retrieve the PIDF-LO being conveyed by the UAC per [RFC3856]. If successful, the PIDF-LO will be returned in the NOTIFY request from the remote host.

A Require header with the geolocation option tag indicates the UAC is requiring the UAS understand this extension or else send an error response. A 420 (Bad Extension) with a geolocation option tag in an Unsupported header would be the appropriate response in this case.

It is possible, but undesirable, for a message to arrive with a body containing a location-by-value, but with no Geolocation header field value pointing to it (potentially no Geolocation header field at all). In this case, the recipient MAY still read and use the message body. Unless stated otherwise by future standards-track publications, a Location-by-reference URI only has meaning within the Geolocation header field and MUST NOT appear in any other SIP header field.

There are 3 Geolocation header parameters,

- o "inserted-by="
- o "used-for-routing"
- o "recipient="

The "inserted-by=" parameter informs a Location Recipient which SIP element added this locationValue to the SIP request. This parameter is mandatory for each locationValue in the request. The value in

the "inserted-by=" parameter is copied into the "inserter="
parameter in each locationErrorValue if there is an error in the

location to be reported back to the location sender. See section 5.2.1.

The "used-for-routing" parameter is included in the locationValue if a SIP server used the location in the request to determine how to route or forward the message towards the ultimate destination. If there are more than one locationValue in the Geolocation header, and it is possible that different locationValues were used to route the message at different times of this request's journey. This is allowed, as it is consistent with the rule that anytime a message is routed based upon a locationValue, a "used-for-routing" parameter is added to the applicable locationValue. This parameter should be present in each locationValue used along the path.

More than one locationValue inserted in a request should be placed the order it was placed, and not rearranged. This informs a Location Recipient which was the last locationValue in the message that was used to route the message. This is for troubleshooting and management reasons.

The "recipient=" header parameter allow recipients to infer the SIP entity type this locationValue is intended to be for. The types are "endpoint", meaning the ultimate destination UAS; "routing-entity", meaning SIP servers; and "both" meaning this locationValue is to be viewed by both types of SIP entities.

Individual header parameters in any received locationValue MUST NOT be modified or deleted in transit to the ultimate destination.

A UAS MUST NOT send location in a response message, as there can be any number of issues/problems with receiving location, and the UAC or proxy servers cannot error a response. Therefore, the UAS, if it wants to send a UAC its location, SHOULD do so in a new request in a separate transaction. This document gives no guidance which SIP request to use.

A UAS MAY include a geolocation option-tag in the Supported header of a response, indicating it does understand this extension, even if location was not in a request to the UAS.

A UAS wishing to dereference a location-by-reference URI contained in a received request will use the 'presence' event package in a SUBSCRIBE request to the URI. If accepted, the PIDF-LO will return to the UAS in a NOTIFY request. If there are any errors during dereferencing, or in the PIDF-LO itself, the UAS will error the original request to the UAC with a locationErrorValue indicating what the UAS concluded was wrong with the location. This is to include any dereferencing problems encountered.

### 5.2.1 UAS Generating a Location Failure Indication

If a received request conveys location, but the UAS has one or more problems with a locationValue in the request (to include while attempting to dereference the UAC's location), the UAS MUST indicate each problem experienced with the location in the request in a 424 (Bad Location Information) response back to the inserting entity if the UAS wants to reject the request because of the location. A Geolocation-Error header is how the UAS informs the UAC of a location-based error within the request. Section 3.4 lists these errors, which are all IANA registered.

Because this extension to SIP allows more than one locationValue in a Geolocation header, each from separate SIP entities, there needs to be a means of identifying which entity inserted a particular locationValue for single error response purposes. This is further complicated because SIP sends a single rejection response, that in this case, needs to go to more than one entity, and be ignored by all other entities not identified in such a way as to not confuse other SIP entities.

Each locationValue has an "inserted-by=" parameter identifying which SIP entity added this locationValue to the request. This value is copied to the locationErrorValue "inserter=" parameter if one needs to be sent, thus identifying the intended target of this locationErrorValue. This locationErrorValue is ignored by all other receivers of this SIP response.

Each locationErrorValue can have more than one error code within it. Each locationErrorValue is destined for one "inserter=" entity. This gives a UAS one mechanism to tell each inserter what the Location Recipient concluded was wrong with what the inserter included (as far as location is concerned). Therefore,

- there MUST be a locationErrorValue for each locationValue that was considered bad by the UAS to ensure each upstream location inserter understands which error code(s)is intended for them (and which to ignore).

- if the PIDF-LO (received by-value or after dereference) contains civic CAtypes that the Location Recipient considers malformed or bad, each CAtype SHOULD be listed in the locationErrorValue to inform the "inserter=" entity what specifically was wrong with the locationValue, in addition to the error code. Without these details, the location inserter might not know what part was malformed or incomplete about the information supplied in the request.

- the CAtype values MUST NOT be sent along with the CAtype names

listed in the locationErrorValue. This is for privacy/security reasons.

- there MUST NOT be more than one locationErrorValue in the response per locationValue in the request.

- there MUST NOT be more than one locationErrorValue in the response for the same locationValue in the request.

- there MUST NOT be a locationErrorValue in the response for a locationValue in the request that was not in error, according to the Location Recipient.

Here is an example of a Geolocation-Error header

```
Geolocation-Error: 106; "node=bob.example.com";
                        "inserter=alice.example.com";
                        CAtype=A3; CAtype=STS;
                        code="incomplete location supplied"
```

See Section 3.4 for further rules about the Geolocation-Error header and the locationErrorValue.

The Geolocation-Error header is permitted in any response. For example, Bob can reply to Alice with a 486 because he's not willing to accept the call at this time, and inform Alice that the location contained in the request was bad in some way. In this case, the 486 would contain a Geolocation-Error header indicating the specific location error experienced

If there is more than one locationValue in a request, and any one of them is valid (i.e., one contains enough information to not generate a 424 if that was the only location present in the request), all other locations MAY be ignored, and a 424 MUST NOT be sent because of these other locations in the request. Another response MAY be sent, which includes a locationErrorValue. This document says nothing about what a Location Recipient does with more than one 'good' location in a request (i.e., which to choose to use).

Further, more than one error code is allowed in the locationErrorValue - each having an "inserter=" parameter. The error codes destined for the same inserter MUST NOT contradict the meaning of the problem the UAS had with a particular locationValue.

A Geolocation-Error is permissible in a 200 OK response. This means everything else in the request was acceptable, but the location was not for a given error code(s). One exception to this set of rules is if a geolocation option-tag was in the Require header in the request. This would necessitate a 424 response.

### 5.3 Proxy Behavior

[RFC3261] states message bodies cannot be added by proxies. However, proxies are permitted to add a header to a request. This

implies that a proxy can add a Geolocation locationValue with location-by-reference URI, but not location-by-value message body. However, if location is already in a SIP request, a SIP server SHOULD NOT add another instance of the UAC's location to the same request. This will likely cause confusion at the Location Recipient as to which to use. This document gives no guidance how a UAS is to deal with more than one location in a SIP request, other than the intended "recipient=" parameter, which has no integrity protection in transit. If more than one locationValue states "recipient=endpoint", this document gives no guidance what the UAS is to do.

A proxy is permitted to read any locationValue, and the associated body, if not S/MIME protected, in transit if present, and MUST use the contents of the header field to make location-based retargeting decisions, if retargeting requests based on location is a function of that proxy.

More than one Geolocation locationValue in a message is permitted, but can cause confusion at the recipient. If a proxy chooses to add a locationValue to a Geolocation header, which would be a local policy decision, the new locationValue MUST be added to the end of the header (after previous locationValue(s)). This is done to create an order of insertion of locationValues along the path. Proxies MUST NOT modify the order of locationValues in a geolocation header.

A proxy wishing to dereference a location-by-reference URI contained in a received request will use the 'presence' event package in a SUBSCRIBE request to the URI. If accepted, the PIDF-LO will return to the proxy in a NOTIFY request. If there are any errors during dereferencing, or in the PIDF-LO itself, the proxy will error the original request to the UAC with a locationErrorValue indicating what the proxy concluded was wrong with the location. This is to include any dereferencing problems encountered.

### 5.3.1 Proxy Behavior with Geolocation Header Parameters

SIP servers MUST NOT delete any existing Geolocation locationValue (URI or header parameter) from a request. A Geolocation locationValue (URI or header parameter) MAY only be modified to by adding a "used-for-routing" parameter to an existing locationValue, if the request was retargeted based on the location within that locationValue. Further modification of this Geolocation header field MUST NOT occur. For example, an existing Geolocation locationValue in a request of:

Geolocation: <cid:alice123@atlanta.example.com>;

```
                inserted-by=alice123@atlanta.example.com;
```

can be modified by a proxy to add the "used-for-routing" parameter,

like this:

```
Geolocation: <cid:alice123@atlanta.example.com>;
              inserted-by=alice123@atlanta.example.com;
              used-for-routing
```

if this is the locationValue the proxy used to make a retargeting decision based upon, but make no other modification.

A SIP server MAY add a new Geolocation locationValue to a SIP request. The proxy SHOULD NOT insert a locationValue of the UAC unless it is reasonably certain it knows the actual location of the endpoint, for example, if it thoroughly understands the topology of the underlying access network and it can identify the device reliably (in the presence of, for example, NAT).

A server adding a locationValue to an existing Geolocation header would look like:

```
Geolocation: <cid:alice123@atlanta.example.com>;
              inserted-by=alice123@atlanta.example.com,
             <sips:3sdefrhy2jj7@lis1.atlanta.example.com>;
              inserted-by=lis1.atlanta.example.com;
```

Notice the locationValue added by the proxy is last among locationValues. This practice MUST be done for all added locationValues.

If this request was then retargeted by an intermediary using the locationValue inserted by the server, the intermediary would add a "used-for-routing" parameter like this:

```
Geolocation: <cid:alice123@atlanta.example.com>;
              inserted-by=alice123@atlanta.example.com,
             <sips:3sdefrhy2jj7@lis1.atlanta.example.com>;
              inserted-by=lis1.atlanta.example.com; used-for-routing
```

It is conceivable that an initial routing decision is made on an one locationValue, and subsequently another routing decision is made on a different locationValue. This retargeting decision can be made on a newly inserted locationValue. While unusual, it can occur. In such a case, proxies MUST NOT remove any existing "used-for-routing" header parameter. In this instance, the SIP server retargeting based on another locationValue MUST add the "used-for-routing" header parameter to the locationValue used for retargeting by this server. This will result in a Geolocation header looking as if it were retargeting more than once, which would be true - and is the desired outcome.

### 5.3.2 Proxy Error Behavior for Sending or Receiving locationErrorValues

For proxies that receive a SIP request that contains a location error, either in a contained message body or after the proxy does a dereference of the LbyR URI, all the rules applicable to a UAS apply here (see Section 5.2.1.), since in this case, the proxy is considered a Location Recipient. Therefore, there is no reason to restate them here, and potentially have the two section be inconsistent. The one thing to add is that a proxy does not need to examine location contained in a request. Section 5.2.1. only applies to proxies that are monitoring or policing location within requests (for whatever reason).

If a proxy inserted a locationValue into a request, it SHOULD be ready to examine the response to that request, in case there is one or more location errors in the response. To a great degree, this scenario has the proxy behaving as a UAC (see section 5.1.1.) that included a locationValue a request, which then receives an error to that locationValue.

If there is one or more locationErrorValues in the response, the proxy SHOULD examine each "inserter=" parameter in each locationErrorValue - looking for one that identifies the proxy. If one matches the proxy's "inserted-by" value, that locationErrorValue is for only that proxy. This locationErrorValue needs to be reviewed for each error code and CAtype contained in the value. The proxy SHOULD attempt to correct for the error reported to it for future insertion of location into requests. This document gives no guidance what the proxy should do to rectify the bad location information, but a future document MAY address this.

## 6. Geopriv Privacy Considerations

Transmitting location information is considered by most to be highly sensitive information, requiring protection from eavesdropping, tracking, and altering in transit. [RFC3693] articulates rules to be followed by any protocol wishing to be considered a Geopriv "Using Protocol", specifying how a transport protocol meetings those rules. This section describes how SIP as a Using Protocol meets those requirements.

Quoting requirement #4 of [RFC3693]:

"The Using Protocol has to obey the privacy and security instructions coded in the Location Object and in the corresponding Rules regarding the transmission and storage of the LO."

This document requires that SIP entities sending or receiving

location MUST obey such instructions.

Quoting requirement #5 of [RFC3693]:

"The Using Protocol will typically facilitate that the keys associated with the credentials are transported to the respective parties, that is, key establishment is the responsibility of the Using Protocol."

[RFC3261] and the documents it references define the key establishment mechanisms.

Quoting requirement #6 of [RFC3693]:

"(Single Message Transfer) In particular, for tracking of small Target devices, the design should allow a single message/packet transmission of location as a complete transaction."

When used for tracking, a simple NOTIFY or UPDATE normally is relatively small, although the PIDF itself can get large. Normal RFC 3261 procedures of reverting to TCP when the MTU size is exceeded would be invoked.

## 7. Security Considerations

Conveyance of physical location of a UAC raises privacy concerns, and depending on use, there probably will be authentication and integrity concerns. This document calls for conveyance to normally be accomplished through secure mechanisms, like S/MIME protecting message bodies (but this is not widely deployed) or TLS protecting the overall signaling. In cases where a session set-up is retargeted based on the location of the UAC initiating the call or SIP MESSAGE, securing the by-value location with an end-to-end mechanism such as S/MIME is problematic, because one or more proxies on the path need the ability to read the location information to retarget the message to the appropriate new destination UAS. Securing the location hop-by-hop, using TLS, protects the message from eavesdropping and modification, but exposes the information to all proxies on the path as well as the endpoint. In most cases, the UAC does not know the identity of the proxy or proxies providing location-based routing services, so that end-to-middle solutions might not be appropriate either.

These same issues exist for basic SIP signaling, but SIP normally does not carry information to physically track a user; making this extension especially sensitive.

When location is inserted by a UAC, which is RECOMMENDED, it can decide whether to reveal its location using hop-by-hop methods. UAC implementations MUST make such capabilities conditional on explicit

user permission, and SHOULD alert a user that location is being conveyed. Proxies inserting location for location-based routing are

unable to meet this requirement, and such use is NOT RECOMMENDED. Proxies conveying location using this extension MUST have the permission of the Target to do so.

One facet within this extension is such that locations can be placed on a remote server, accessible with the possession of a URI. The concept of a location-by-reference URI has its own security considerations. It is tempting to assume that the dereference would have authentication, authorization and other security mechanisms that limit the access to information. Unfortunately, this might not be true. The access network the UAC is connected to can be the source of location reference, and it might not have any credentialing mechanism suitable for controlling access to location. Consider, specifically, a nomadic user connected to an access network in a hotel. The UAC has no way to provide a credential acceptable to the hotel Location Server (LS) to any of its intended Location Recipients. The recipient of a reference does not know if a reference has appropriate authorization policies or not. The LS should provide location to any requestor.

Accordingly, possession of the reference should be considered equivalent to possession of the value, and the reference should be treated with the same degree of care as the value. Specifically, TLS MUST be used to protect the security of the reference. Notice that this does not constrain the dereference protocol to use TLS. That specification is left entirely to the dereferencing protocol

Because SIP servers can add location in transit, made more easy of the server is a Session Border Controller or B2BUA, this might cause there to be conflicting location information (error-code=6), which could be purposeful to error the request or just cause operation problems. This problem might be inadvertent, compounded by the fact that there will likely be some SIP servers that add location on every call set-up.

There is no integrity on any locationValue or locationErrorValue header parameter, so recipients of either header need to implicitly trust the header contents, and take whatever precautions each entity deems appropriate give these facts.

## 8. IANA Considerations

The following are the IANA considerations made by this SIP extension. Modifications and additions to these registrations require a standards track RFC (Standards Action).

### 8.1 IANA Registration for the SIP Geolocation Header

The SIP Geolocation header is created by this document, with its

definition and rules in Section 3.2 of this document, to be added to the sip-parameters.

The Geolocation Header has the following header parameters to be Registered in a new table:

Geolocation Header parameters

| Header Parameters | Parameter-values | Reference |
|---|---|---|
| recipient | endpoint | RFC XXXX (this document) |
| recipient | routing-entity | RFC XXXX (this document) |
| recipient | both | RFC XXXX (this document) |

## 8.2 IANA Registration for New SIP Option Tag

The SIP option-tag "geolocation" is created by this document, with the definition and rule in Section 3.5 of this document, to be added to sip-parameters within IANA.

## 8.3 IANA Registration for Response Code 424

Reference: RFC-XXXX (i.e., this document)
Response code: 424 (recommended number to assign)
Default reason phrase: Bad Location Information

This SIP Response code is defined in section 3.3 of this document.

## 8.4 IANA Registration of New Geolocation-Error Header

The SIP Geolocation-error header is created by this document, with its definition and rules in Section 3.4 of this document, to be added to the sip-parameters.

## 8.5 IANA Registration for the SIP Geolocation-Error Codes

New location specific Geolocation-Error codes are created by this document, and registered in a new table at sip-parameters within IANA. Details of these error codes are in Section 3.4 of this document.

Geolocation-Error codes
---
Geolocation-Error codes provide reason for the error discovered by Location Recipients, to place into SIP response messages to inform the location inserter of the error.

| Code | Description | Reference |
|---|---|---|

| | | |
|---|---|---|
| 100 | Location Format Not Supported: the location format | [this doc] |

| | | |
|---|---|---|
| | supplied in the request, by-value or by-reference, was not supported. | |
| 101 | Coordinate-location Format Desired: the location format supplied in the request was understood and supported, but that the recipient, or an application on the recipient, can or prefers to only process location in the coordinate-location format. | [this doc] |
| 102 | Civic-location Format Desired: the location format supplied in the request was understood and supported, but that the recipient, or an application on the recipient, can or prefers to only process location in the civic-location format. | [this doc] |
| 103 | Cannot Parse Location Supplied: the location provided, whether by-value or by-reference, in a request is not well formed. | [this doc] |
| 104 | Cannot Find Location: the location was expected in the request, but the recipient cannot find it. | [this doc] |
| 105 | Conflicting Locations Supplied: a Location Recipient received more than one location describing where the Target is, and is either unsure which whole location is true or which parts of multiple locations make up where the Target is. | [this doc] |
| 106 | Incomplete Location Supplied: there is not enough location information in the request to determine where the location Target is. | [this doc] |
| 107 | Cannot Dereference: the act of dereferencing failed to return the Target's location. This generally means the supplied URI is bad. | [this doc] |
| 108 | Dereference Denied: there was insufficient authorization to dereference the Target's location. | [this doc] |
| 109 | Dereference Timeout: the dereferencing node has not received the Target's location within a reasonable. timeframe | [this doc] |
| 110 | Cannot Process Dereference: the dereference protocol has received an overload condition error, indicating the location cannot be accessed at this time. | [this doc] |
| 120 | Unsupported Scheme - sip desired: the location | [this doc] |

dereferencer cannot dereference using the
location-by-reference URI scheme supplied, and

prefers a sip-uri.

121 Unsupported Scheme - sips desired: the location dereferencer cannot dereference using the location-by-reference URI scheme supplied, and prefers a sips-uri. [this doc]

122 Unsupported Scheme - pres desired: the location dereferencer cannot dereference using the location-by-reference URI scheme supplied, and prefers a pres-uri. [this doc]

## 9. Acknowledgements

To Dave Oran for helping to shape this idea. To Jon Peterson and Dean Willis on guidance of the effort. To Allison Mankin, Dick Knight, Hannes Tschofenig, Henning Schulzrinne, James Winterbottom, Jeroen van Bemmel, Jean-Francois Mule, Jonathan Rosenberg, Keith Drage, Marc Linsner, Martin Thomson, Mike Hammer, Paul Kyzivat, Shida Shubert, Umesh Sharma, Richard Barnes, Ted Hardie and Matt Lepinski for constructive feedback. A special thanks to Dan Wing for help with the S/MIME example, and to Robert Sparks for many helpful comments and the proper building of the Geolocation-Error header.

## 10. References

### 10.1 References - Normative

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

[RFC4119] J. Peterson, "A Presence-based GEOPRIV Location Object Format", RFC 4119, December 2005

[RFC2119] S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March 1997

[RFC2392] E. Levinson, " Content-ID and Message-ID Uniform Resource Locators", RFC 2393, August 1998

[RFC3863] H. Sugano, S. Fujimoto, G. Klyne, A. Bateman, W. Carr, J. Peterson, "Presence Information Data Format (PIDF)", RFC 3863, August 2004

[RFC3856] J. Rosenberg, " A Presence Event Package for the Session Initiation Protocol (SIP)", RFC 3856, August 2004

[RFC3859] J. Peterson, "Common Profile for Presence (CPP)", RFC 3859, August 2004

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

[RFC3311] J. Rosenberg, "The Session Initiation Protocol (SIP) UPDATE Method", RFC 3311, October 2002

[RFC3265] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002.

[RFC3262] Rosenberg, J. and H. Schulzrinne, "Reliability of Provisional Responses in Session Initiation Protocol (SIP)", RFC 3262, June 2002.

[IANA-civic] http://www.iana.org/assignments/civic-address-types-registry

## 10.2 References - Informative

[RFC3693] J. Cuellar, J. Morris, D. Mulligan, J. Peterson. J. Polk, "Geopriv Requirements", RFC 3693, February 2004

[RFC3825] J. Polk, J. Schnizlein, M. Linsner, "Dynamic Host Configuration Protocol Option for Coordinate-based Location Configuration Information", RFC 3825, July 2004

[RFC4776] H. Schulzrinne, " Dynamic Host Configuration Protocol (DHCPv4 and DHCPv6) Option for Civic Addresses Configuration Information ", draft-ietf-geopriv-dhcp-civil-09, "work in progress", January 2006

Author Information

James Polk
Cisco Systems
3913 Treemont Circle 33.00111N
Colleyville, Texas 76034 96.68142W

Phone: +1-817-271-3552
Email: jmpolk@cisco.com

Brian Rosen
NeuStar, Inc.

470 Conrad Dr.  40.70497N
Mars, PA  16046  80.01252W

US

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

## Appendix A. Requirements for SIP Location Conveyance

The following subsections address the requirements placed on the UAC, the UAS, as well as SIP proxies when conveying location. There is a motivational statement below each requirements that is not obvious in intent

### A.1 Requirements for a UAC Conveying Location

UAC-1 The SIP INVITE Method [RFC3261] must support location conveyance.

UAC-2 The SIP MESSAGE method [RFC3428] must support location conveyance.

UAC-3 SIP Requests within a dialog should support location conveyance.

UAC-4 Other SIP Requests may support location conveyance.

UAC-5 There must be one, mandatory to implement means of transmitting location confidentially.

Motivation: interoperability

UAC-6 It must be possible for a UAC to update location conveyed at any time in a dialog, including during dialog establishment.

Motivation: in case a UAC has moved prior to the establishment of a dialog between UAs, the UAC must be able to send new location information. In the case of location having been conveyed, and the UA moves, it needs a means to update the conveyed to party of this location change.

UAC-7 The privacy and security rules established within [RFC3693] that would categorize SIP as a 'Using Protocol' must be met.

UAC-8 The PIDF-LO [RFC 4119] is a mandatory to implement format for location conveyance within SIP, whether included by-value or by-reference.

Motivation: interoperability with other IETF location protocols and mechanisms

UAC-9 There must be a mechanism for the UAC to request the UAS send

its location

UAC-9 has been DEPRECATED by the SIP WG, due to the many problems this requirement would have caused if implemented. The solution is for the above UAS to send a new request to the original UAC with the UAS's location.

UAC-10 There must be a mechanism to differentiate the ability of the UAC to convey location from the UACs lack of knowledge of its location

Motivation: Failure to receive location when it is expected can be because the UAC does not implement this extension, or it can be that the UAC implements the extension, but does not know where it is. This may be, for example, due to the failure of the access network to provide a location acquisition mechanisms the UAC understands. These cases must be differentiated.

UAC-11 It must be possible to convey location to proxy servers along the path.

Motivation: Location-based routing.

## A.2 Requirements for a UAS Receiving Location

The following are the requirements for location conveyance by a UAS:

UAS-1 SIP Responses must support location conveyance.

Just as with UAC-9, UAS-1 has been DEPRECATED by the SIP WG, due to the many problems this requirement would have caused if implemented. The solution is for the above UAS to send a new request to the original UAC with the UAS's location.

UAS-2 There must be a unique 4XX response informing the UAC it did not provide applicable location information.

In addition, requirements UAC-5, 6, 7 and 8 apply to the UAS

## A.3 Requirements for SIP Proxies and Intermediaries

The following are the requirements for location conveyance by a SIP proxies and intermediaries:

Proxy-1 Proxy servers must be capable of adding a Location header field during processing of SIP requests.

Motivation: Provide the capability of network assertion of location

when UACs are unable to do so, or when network assertion is more reliable than UAC assertion of location

Note: Because UACs connected to sip signaling networks may have widely varying access network arrangements, including VPN tunnels and roaming mechanisms, it may be difficult for a network to reliably know the location of the endpoint. Proxy assertion of location is NOT RECOMMENDED unless the sip signaling network has reliable knowledge of the actual location of the Targets.

Proxy-2 There must be a unique 4XX response informing the UAC it did not provide applicable location information.

## Appendix B. Example of INVITE with S/MIME encrypted Civic PIDF-LO

This appendix gives an *EXAMPLE* (meaning this might contain errors based on future review) of a SIP INVITE request that points to the same position on the earth as the coordinate based example that's in section 4.1 in the body of this document:

The INVITE request is TLS hop-by-hop encrypted, and the location-by-value message body is S/MIME encrypted. This example shows the location message body in its unencrypted form for clarity. The message body lines below that have the '$' signs are S/MIME encrypted. In this example, the SDP is not S/MIME encrypted.

```
INVITE sips:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/TLS pc33.atlanta.example.com
  ;branch=z9hG4bK74bf9
Max-Forwards: 70
To: Bob <sips:bob@biloxi.example.com>
From: Alice <sips:alice@atlanta.example.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@atlanta.example.com
Geolocation: <cid:alice123@atlanta.example.com>
  ;inserted-by=alice@atlanta.example.com ;recipient=endpoint
Supported: geolocation
Accept: application/sdp, application/pidf+xml
CSeq: 31862 INVITE
Contact: <sips:alice@pc33.atlanta.example.com>
Content-Type: multipart/mixed; boundary=boundary1
Content-Length: ...

--boundary1

Content-Type: application/sdp

...SDP goes here

--boundary1
```

```
   Content-Type: application/pkcs7-mime;
```

```
      smime-type=enveloped-data; name=smime.p7m
   Content-ID: alice123@atlanta.example.com

$  Content-Type: application/pidf+xml
$
$  <?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:cl="urn:ietf:params:xml:ns:pidf:geopriv10:civicAddr"
$         entity="pres:alice@atlanta.example.com">
$       <tuple id="sg89ae">
$        <timestamp>2007-07-09T14:00:00Z</timestamp>
$        <status>
$         <gp:geopriv>
$           <gp:location-info>
$             <cl:civicAddress>
$               <cl:country>US</cl:country>
$               <cl:A1>Texas</cl:A1>
$               <cl:A3>Colleyville</cl:A3>
$               <cl:HNO>3913</cl:HNO>
$               <cl:A6>Treemont</cl:A6>
$               <cl:STS>Circle</cl:STS>
$               <cl:PC>76034</cl:PC>
$               <cl:NAM>Haley's Place</cl:NAM>
$               <cl:FLR>1</cl:FLR>
$             <cl:civicAddress>
$           </gp:location-info>
$           <gp:usage-rules>
$             <gp:retransmission-allowed>no</gp:retransmission-allowed>
$             <gp:retention-expiry>2007-07-27T18:00:00Z</gp:retention-
$                           expiry>
$           </gp:usage-rules>
$           <gp:method>DHCP</gp:method>
$           <gp:provided-by>www.example.com</gp:provided-by>
$         </gp:geopriv>
$        </status>
$       </tuple>
$      </presence>
   --boundary1--
```

## Full Copyright Statement

## Intellectual Property



## Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).